Document Title	Specification of Network Management
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	898

Document Status	Final
Part of AUTOSAR Standard	Adaptive Platform
Part of Standard Release	18-10

Document Change History

Date	Release	Changed by	Description
2018-10-30	18-10	AUTOSAR Release Management	• Updated interaction with State Management • Removed APIs and Services (interaction is done via SM) • Temporary removed user data access to applications
2018-03-29	18-03	AUTOSAR Release Management	• Initial relase

Disclaimer

Table of Contents

1 Introduction and functional overview

This specification describes the functionality, API and the configuration of the `Network Management` for the AUTOSAR `Adaptive Platform`.

Adaptive Network Management (NM) is intended to work independent of the communication stack used. Its main purpose is to coordinate the transition between normal operation and bus-sleep mode of the underlying networks (physical and partial networks).

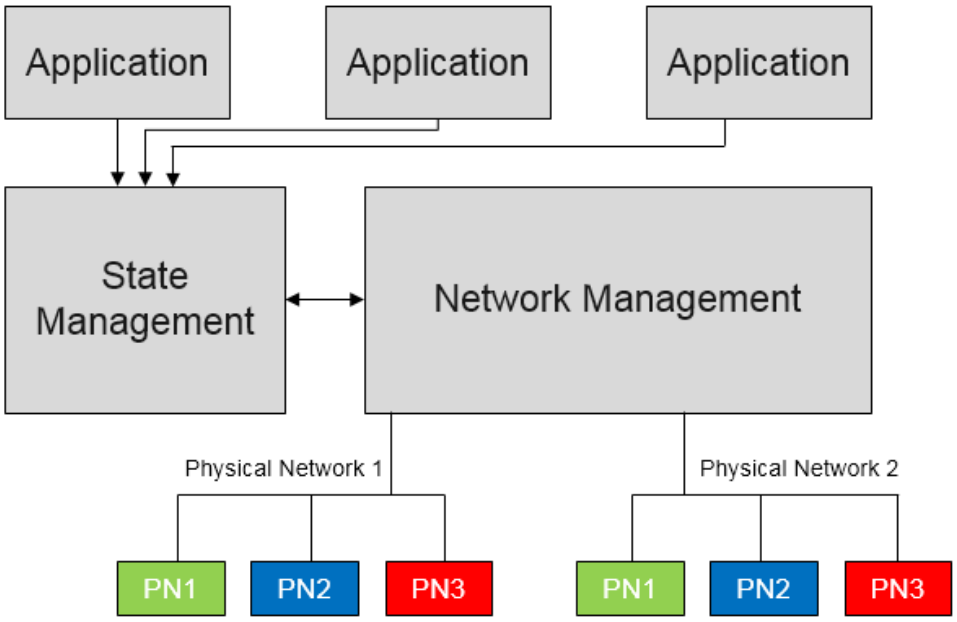

Figure 1.1: Architecture overview with example applications

2 Acronyms and Abbreviations

The glossary below includes acronyms and abbreviations relevant to the Adaptive Network Management that are not included in the AUTOSAR glossary [1].

Abbreviation / Acronym:	Description:
API	Application Programming Interface
CBV	Control Bit Vector
CM	Communication Management
CWU	Car Wakeup
EM	Execution Management
IP	Internet Protocol
MTU	Maximum Transmission Unit
NM	Network Management
NM Node	A node that supports network management. Please note that network node, node and NM node are used with the same meaning througout the document.
PN	Partial Network
PNI	Partial Network Information
UDP	User Datagram Protocol

Terms:	Description:
Bus communication	Communication on the physical medium
Logical Network	A network in which devices can be addressed independent from the actual network technology.
NM cluster	Set of NM nodes coordinated with the use of the NM algorithm.
NM message	Refers to the payload transmitted in a packet. It contains the NM User Data, Partial Network Information as well as the Control Bit Vector and the Source Node Identifier.
NM packet	Refers to an Ethernet Frame containing an IP as well as an UDP header in addition to a NM message. Please note that adaptive network management is currently only supported for Ethernet.
PN communication	Communication during partial network operation
Physical channel	A channel enabling communication using physical devices, such as I/O ports and cables.
Repeat Message Request Bit Indication	Repeat Message Bit set in the Control Bit Vector of a received NM message.

3 Related documentation

3.1 Input documents

AUTOSAR provides a General Specification on Basic Software modules [2, RS General], which is also valid for the NM.

Thus, the specification SWS BSW General shall be considered as additional and required specification for NM.

[1] Glossary
AUTOSAR_TR_Glossary

[2] General Requirements specific to Adaptive Platform
AUTOSAR_RS_General

[3] Specification of the AUTOSAR Network Management Protocol
AUTOSAR_PRS_NetworkManagementProtocol

[4] Requirements on AUTOSAR Network Management
AUTOSAR_RS_NetworkManagement

3.2 Related specification

N/A

4 Constraints and assumptions

4.1 Limitations

The Adaptive Network Management is actually only supporting UdpNM.

The Adaptive Network Management does not allow node detection (Repeat Message State) but only handles incoming requests.

The Adaptive Network Management cannot be configured as the master network coordinator.

The Adaptive Network Management does not support coordinated shutdown using the information in CBV.

The Adaptive Network Management does not support passive mode and passive start-up. Passive start-up would mean that a node has started (i.e. goes to Normal mode), but the network has been woken up by another node.

Modeling part for mapping the logical networks to the BitVector positions as defined in chapter 7.3 is not available in the manifest.

Update and access of User Data was removed as the service interface to Applications has been removed. State Management will control the network request/release and it must be clarified if user data changes/indications shall be done via State Management or directly by applications.

4.2 Applicability to car domains

AUTOSAR Adaptive Network Management can be used for all car domains.

5 Dependencies to other modules

There are no dependencies to other functional clusters.

5.1 Platform dependencies

This specification is part of the AUTOSAR Adaptive Platform and therefore depends on it.

5.2 Protocol layer dependencies

The Adaptive Network Management is based on the protocol mentioned in PRS NetworkManagementProtocol [3].

Adaptive Network Management uses functionality of the underlying communication stack in order to send or receive NM messages on the physical networks.

6 Requirements Tracing

The following table references the requirements specified in RS Adaptive Network Management [4] and links to the fulfillment of these. Please note that if column “Satisfied by” is empty for a specific requirement, or a requirement is not reported in the table, it means that this requirement is not fulfilled by this document.

Requirement	Description	Satisfied by
[RS_Nm_00044]	The NM shall be applicable to different types of communication systems which are in the scope of Autosar and support a bus sleep mode.	[SWS_ANM_00001] [SWS_ANM_00004] [SWS_ANM_00005] [SWS_ANM_00006] [SWS_ANM_00007] [SWS_ANM_00008] [SWS_ANM_00009] [SWS_ANM_00012] [SWS_ANM_00013] [SWS_ANM_00016] [SWS_ANM_00017] [SWS_ANM_00021] [SWS_ANM_00044] [SWS_ANM_00046] [SWS_ANM_00047] [SWS_ANM_00062] [SWS_ANM_00070]
[RS_Nm_00047]	NM shall provide a service to request to keep the bus awake and a service to cancel this request.	[SWS_ANM_00011] [SWS_ANM_00014] [SWS_ANM_00015] [SWS_ANM_00016] [SWS_ANM_00018] [SWS_ANM_00019] [SWS_ANM_00020] [SWS_ANM_00022] [SWS_ANM_00023] [SWS_ANM_00025]
[RS_Nm_00048]	NM shall put the communication controller into sleep mode if there is no bus communication	[SWS_ANM_00024]
[RS_Nm_00050]	The NM shall provide the current state of NM	[SWS_ANM_00031] [SWS_ANM_00063]
[RS_Nm_00054]	There shall be a deterministic time from the point where all nodes agree to go to bus sleep to the point where bus is switched off.	[SWS_ANM_00024]

Requirement	Description	Satisfied by
[RS_Nm_00150]	Specific functions of the Network Management shall be configurable	[SWS_ANM_00007] [SWS_ANM_00013] [SWS_ANM_00028] [SWS_ANM_00033] [SWS_ANM_00035] [SWS_ANM_00040] [SWS_ANM_00044] [SWS_ANM_00047] [SWS_ANM_00051] [SWS_ANM_00077] [SWS_ANM_00081]
[RS_Nm_00151]	The Network Management algorithm shall allow any node to integrate into an already running NM cluster	[SWS_ANM_00037] [SWS_ANM_00038] [SWS_ANM_00071]
[RS_Nm_02503]	The NM API shall optionally give the possibility to send user data	[SWS_ANM_00039]
[RS_Nm_02504]	The NM API shall optionally give the possibility to get user data	[SWS_ANM_00039] [SWS_ANM_00048]
[RS_Nm_02505]	The NM shall optionally set the local node identifier to the NM-message	[SWS_ANM_00033] [SWS_ANM_00034]
[RS_Nm_02508]	Every node shall have associated with it a node identifier that is unique in the NM-cluster	[SWS_ANM_00034]
[RS_Nm_02517]	<Bus>Nm shall support Partial Networking on CAN, FlexRay and Ethernet	[SWS_ANM_00067] [SWS_ANM_00068]
[RS_Nm_02519]	The NM Control Bit Vector shall contain a PNI (Partial Network Information) bit.	[SWS_ANM_00067] [SWS_ANM_00068]
[RS_Nm_02520]	<Bus>Nm shall evaluate the PNI bit in the NM message	[SWS_ANM_00055] [SWS_ANM_00067] [SWS_ANM_00068]
[RS_Nm_02522]	<Bus>Nm shall calculate the combined partial network request status EIRA	[SWS_ANM_00066]

7 Functional specification

The Adaptive Network Management offers services that allow the applications to request the necessary network states for logical networks that can be mapped to physical or partial networks.

To do so, the following functionalities are provided:

1) Field for requesting and releasing logical networks
2) Support for partial networking

7.1 Architectural Overview

Figure 7.1 gives an overview of the Adaptive NM service.

The following figure shows an overview on the interaction between `State Management` and `Network Management` as well as an example mapping between logical networks, partial networks and physical networks.

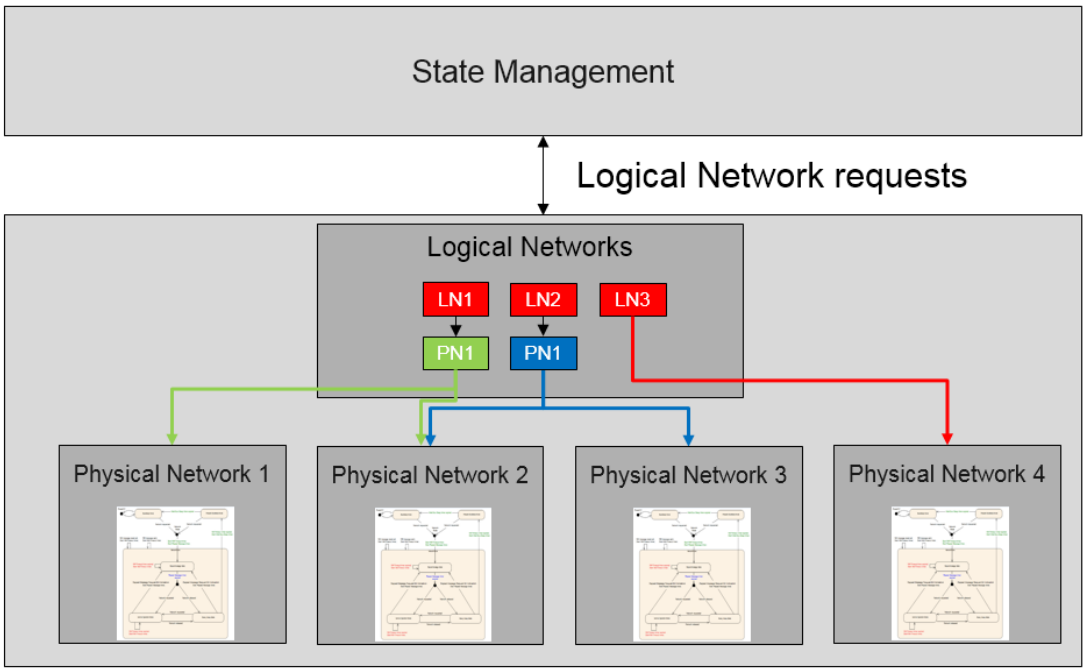

Figure 7.1: Overview Of Network Management

7.2 Network Management Algorithm

The AUTOSAR Adaptive NM is based on decentralized direct network management strategy, which means that every network node performs activities self-sufficient depending only on the NM packets received and/or transmitted within the communication system.

The AUTOSAR Adaptive NM algorithm is based on periodic NM packets, which are received by all nodes in the cluster via multicast. Reception of NM packets indicates that sending nodes want to keep the NM-cluster awake. If any node is ready to go to sleep mode, it stops sending NM packets, but as long as NM packets from other nodes are received, it postpones transition to sleep mode. Finally, if a dedicated timer elapses because no NM packets are received anymore, every node initiates transition to the sleep mode.

If any node in the NM-cluster requires bus-communication, it can keep the NM-cluster awake by transmitting NM packets.

The main concept of the AUTOSAR Adaptive NM coordination algorithm can be defined by the following key-requirement.

[SWS_ANM_00001] ⌈ Every network node shall transmit periodic NM messages as long as it requires bus-communication; otherwise it shall not transmit NM messages. ⌋ *(RS_Nm_00044)*

7.3 NetworkControl

Logical networks are a mapped to one partial or physical network, while a partial networks can be mapped to one or multiple physical networks.

With the introduction of the `State Management` functional cluster, `Network Management` no longer receives logical network requests from applications, instead they are controlled by the `State Management`. `State Management` can split the one or more applications in multiple functions that might require network communication. Applications (or part of) would then request different functions to be activated/deactivated from `State Management` and then `State Management` would in turn, depending on configuration, request/release different logical networks. The end effect being that one or multiple physical networks are activated/deactivated.

[SWS_ANM_00063] ⌈Each port offered by `NM` shall enable control of single logical network. ⌋*(RS_Nm_00050)*

[SWS_ANM_00066] ⌈Each port shall be mapped over the manifest to the logical networks. ⌋*(RS_Nm_02522)*

[SWS_ANM_00067] ⌈In case of partial networking a mapping between partial network(s) and physical network(s) needs to be configured. ⌋*(RS_Nm_02517, RS_Nm_02519, RS_Nm_02520)*

[SWS_ANM_00068] ⌈In case of no partial networking, the mapping will be done between port and physical network(s). ⌋*(RS_Nm_02517, RS_Nm_02519, RS_Nm_02520)*

7.4 Operational Modes

This chapter describes the operational modes of the AUTOSAR Adaptive NM.

[SWS_ANM_00062] ⌈ Each NM Service instance shall realize the state machine mentioned in the figure described below.⌋ *(RS_Nm_00044)* (Figure 7.2)

Note: The state machine in Figure 7.2 is applied to physical channels. In case of partial networking, the Nm module should additionally take care of relevant PNs.

[SWS_ANM_00004] ⌈ The AUTOSAR Adaptive NM shall contain three operational modes:

- Network Mode
- Prepare Bus-Sleep Mode
- Bus-Sleep Mode

⌋ *(RS_Nm_00044)*

For more information check the following chapters:

- Network Mode, see 7.4.1
- Prepare Bus-Sleep Mode, see 7.4.2
- Bus-Sleep Mode, see 7.4.3

These modes will not be visible to the Adaptive Application as it is.

When the NM is in Network mode it implies that the network is requested or active. And the logical network information bit will be set to 1.

When the NM is in Prepare Bus-Sleep or Bus-Sleep Mode, It implies that the network is released or inactive. And the logical network information bit will be set to 0.

The following figure shows the state diagram. Mode change related transitions are denoted in green and error handling related transmissions in red.

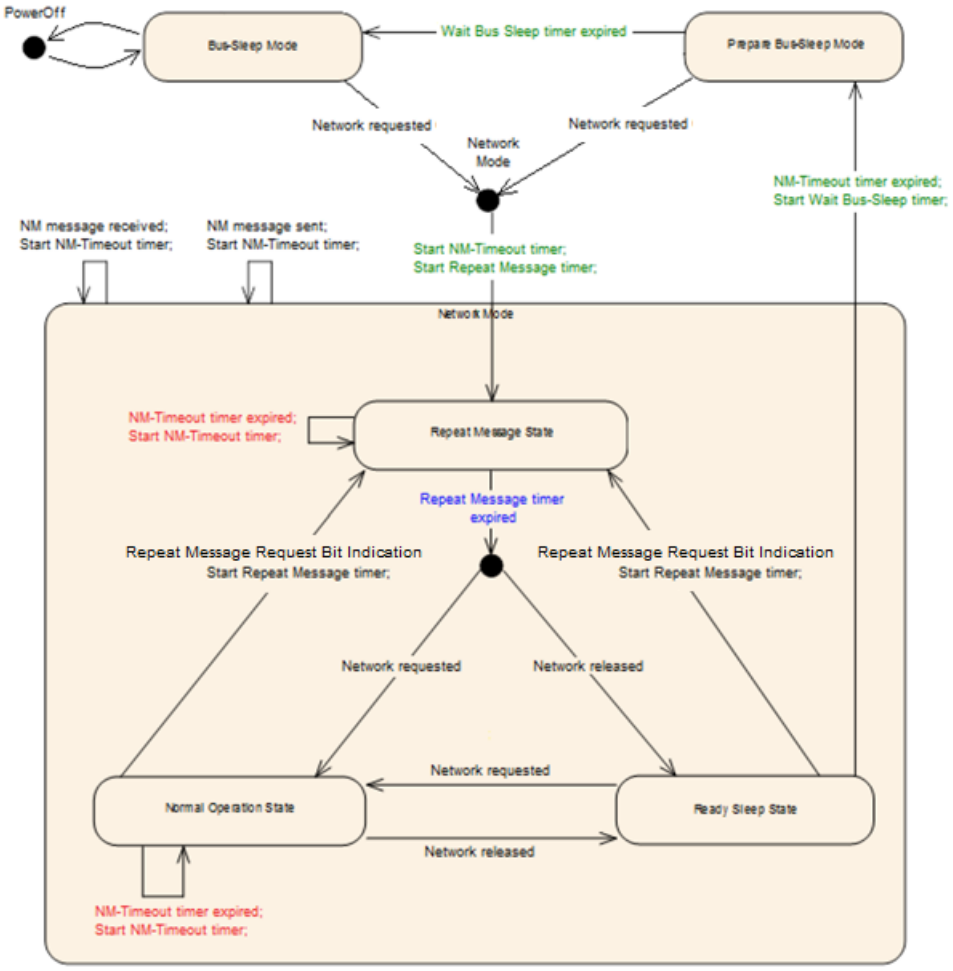

Figure 7.2: State Chart Diagram

7.4.1 Network Mode

[SWS_ANM_00005] ⌈ The Network Mode shall consist of three internal states:

- Repeat Message State
- Normal Operation State
- Ready Sleep State

⌋*(RS_Nm_00044)*

For more information check the following chapters:

- Repeat Message State, see 7.4.1.1
- Normal Operation State, see 7.4.1.2
- Ready Sleep State, see 7.4.1.3

[SWS_ANM_00006] ⌈ When the Network Mode is entered from Bus-Sleep Mode or Prepare Bus-Sleep Mode, by default, the Repeat Message State shall be entered. ⌋ *(RS_Nm_00044)*

[SWS_ANM_00007] ⌈ When the Network Mode is entered, the NM-Timeout Timer shall be started with the value `NmTimeoutTime`. ⌋*(RS_Nm_00044, RS_Nm_00150)*

[SWS_ANM_00008] ⌈ Upon successful reception of a NM message in Network Mode, the NM-Timeout Timer shall be restarted with the value NmTimeoutTime. ⌋ *(RS_Nm_00044)*

[SWS_ANM_00009] ⌈ Upon successful transmission of a NM message in Network Mode, the NM-Timeout Timer shall be restarted with the value NmTimeoutTime. ⌋ *(RS_Nm_00044)*

7.4.1.1 Repeat Message State

The Repeat Message State ensures, that any transition from Bus-Sleep or Prepare Bus-Sleep to the Network Mode becomes visible for the other nodes on the network. Additionally, it ensures that any node stays active for a minimum amount of time.

[SWS_ANM_00011] ⌈ When the Repeat Message State of Network Mode is entered from Bus-Sleep Mode, Prepare-Bus-Sleep Mode or from within Network Mode (Normal Operation State or Ready Sleep State) transmission of NM messages shall be (re-) started. ⌋*(RS_Nm_00047)*

[SWS_ANM_00012] ⌈ When the NM-Timeout Timer expires in the Repeat Message State, the NM-Timeout Timer shall be restarted. ⌋*(RS_Nm_00044)*

[SWS_ANM_00013] ⌈ The NM shall stay in the Repeat Message State for a configurable amount of time determined by the `NmRepeatMessageTime`; after that time the Repeat Message State shall be left. ⌋*(RS_Nm_00044, RS_Nm_00150)*

[SWS_ANM_00014] ⌈ When Repeat Message State is left, the Normal Operation State shall be entered, if the network has been requested. ⌋*(RS_Nm_00047)*

[SWS_ANM_00015] ⌈ When Repeat Message State is left, the Ready Sleep State shall be entered, if the network has been released. ⌋*(RS_Nm_00047)*

[SWS_ANM_00070] ⌈ The Repeat Message State of Network Mode is entered from Bus-Sleep Mode or Prepare-Bus-Sleep Mode by default, when a logical network is requested using RequestNetwork method and the NM module shall transmit a NM message immediately.

⌋*(RS_Nm_00044)*

7.4.1.2 Normal Operation State

The Normal Operation State ensures that any node can keep the NM-cluster awake as long as the network functionality is required.

[SWS_ANM_00016] ⌈ When the Normal Operation State is entered from Ready Sleep State, transmission of NM messages shall be started immediately. ⌋*(RS_Nm_00047, RS_Nm_00044)*

[SWS_ANM_00017] ⌈ When the NM-Timeout Timer expires in the Normal Operation State, the NM-Timeout Timer shall be restarted. ⌋*(RS_Nm_00044)*

[SWS_ANM_00018] ⌈ When the network is released and the current state is Normal Operation State, the Normal Operation State shall be left and the Ready Sleep state shall be entered. ⌋*(RS_Nm_00047)*

[SWS_ANM_00019] ⌈ If Repeat Message Request Bit (set in the CBV of the received NM message) is received in the Normal Operation State, the Normal Operation State shall be left and the Repeat Message State shall be entered. ⌋*(RS_Nm_00047)*

7.4.1.3 Ready Sleep State

The Ready Sleep State ensures that any node in the NM-cluster waits with the transition to the Prepare Bus-Sleep Mode as long as any other node keeps the NM-cluster awake.

[SWS_ANM_00020] ⌈ When the Ready Sleep State is entered from Repeat Message State or Normal Operation State, transmission of NM messages shall be stopped. ⌋ *(RS_Nm_00047)*

[SWS_ANM_00021] ⌈ When the NM-Timeout Timer expires in the Ready Sleep State, the Ready Sleep State shall be left and the Prepare Bus-Sleep Mode shall be entered. ⌋*(RS_Nm_00044)*

[SWS_ANM_00022] ⌈ When the network is requested and the current state is the Ready Sleep State, the Ready Sleep State shall be left and the Normal Operation State shall be entered. ⌋*(RS_Nm_00047)*

[SWS_ANM_00023] ⌈ If Repeat Message Request Bit (set in the CBV of the received NM message) is received in the Ready Sleep State, the Ready Sleep State shall be left and the Repeat Message State shall be entered. ⌋*(RS_Nm_00047)*

Note: Handling of multiple transition conditions which might arise at the same time (e.g. NM-Timeout timer expires vs. network is requested) is considered to be implementation-specific.

7.4.2 Prepare Bus-Sleep Mode

The purpose of the Prepare Bus Sleep state is to ensure that all nodes have time to stop their network activity before the Bus Sleep state is entered. Bus activity is calmed down (i.e. queued messages are transmitted in order to empty all TX-buffers) and finally there is no activity on the bus in the Prepare Bus-Sleep Mode.

[SWS_ANM_00024] ⌈ The NM shall stay in the Prepare Bus-Sleep Mode for an amount of time determined by the `NmWaitBusSleepTime`; after that time, the Prepare Bus-Sleep Mode shall be left and the Bus-Sleep Mode shall be entered. ⌋*(RS_Nm_00048, RS_Nm_00054)*

[SWS_ANM_00025] ⌈ Upon successful reception of a NM message in the Prepare Bus-Sleep Mode, the Prepare Bus-Sleep Mode shall be left and the Network Mode shall be entered; by default, the Repeat Message State is entered. ⌋*(RS_Nm_00047)*

Rationale: Other nodes in the cluster are still in Prepare Bus-Sleep Mode; in the exceptional situation described above, transition into the Bus-Sleep Mode shall be avoided and bus-communication shall be restored as fast as possible.

7.4.3 Bus-Sleep Mode

The purpose of the Bus-Sleep state is to reduce power consumption in the node, when no messages are to be exchanged. Transmission and reception capabilities can be switched off if supported by hardware.

If a configurable amount of time determined by `NmTimeoutTime` + `NmWaitBusSleepTime` is identically configured for all nodes in the network management cluster, all nodes in the network management cluster that are coordinated with use of the AUTOSAR NM algorithm perform the transition into the Bus-Sleep Mode at approximately the same time.

[SWS_ANM_00028] ⌈ If bus communication is released and there are no NM messages on the bus for a configurable amount of time, determined by `NmTimeoutTime` + `NmWaitBusSleepTime`, transition into the Bus-Sleep Mode shall be performed from the Network Mode. ⌋*(RS_Nm_00150)*

[SWS_ANM_00031] ⌈ In the Bus-Sleep Mode the corresponding bit of `NetworkState BitVector`, which represents the relevant logical network, shall be 0. ⌋ *(RS_Nm_00050)*

Note: In Bus-Sleep Mode, it is assumed that all nodes in a cluster are in this state. Typically, all nodes request the communication approximately at the same time by a common trigger, for instance a wake-up line.

7.5 Message Format

Message Layout is shown in [3], chapter 5.1.

Note: As mentioned in [3], the length of an NM packet shall not exceed the MTU (Maximum Transmission Unit) of the underlying physical transport layer.

7.5.1 Source Node Identifier

[SWS_ANM_00033] ⌈ The location of the source node identifier shall be configurable using `NmPduNidPosition`, to position Byte 0 or Byte 1 or Off (default: Byte 0). ⌋ *(RS_Nm_00150, RS_Nm_02505)*

[SWS_ANM_00034] ⌈ The source node identifier shall be set with configurable NodeId value `NmNodeId` unless the location of the source node identifier is set to Off. ⌋ *(RS_Nm_02508, RS_Nm_02505)*

7.5.2 Control Bit Vector

The format (bit-layout) and definition of the CBV is shown in [3], chapter 5.1.2 Control Bit Vector.

[SWS_ANM_00035] ⌈ The location of the Control Bit Vector shall be configurable using `NmPduCbvPosition`, to position Byte 0 or Byte 1 or Off (default: Byte 1). ⌋ *(RS_Nm_00150)*

[SWS_ANM_00037] ⌈ Repeat Message Request Bit shall always be set to 0 in the transmitted NM message. ⌋*(RS_Nm_00151)*

[SWS_ANM_00038] ⌈ Active Wakeup Bit shall always be set to 0 in the transmitted NM message. ⌋*(RS_Nm_00151)*

[SWS_ANM_00071] ⌈ NM Coordinator Sleep Ready Bit shall always be set to 0 in the transmitted NM message. ⌋*(RS_Nm_00151)*

7.5.3 User Data

[SWS_ANM_00039] ⌈ It shall be possible to enable or disable the support of NM user data, using parameter `NmUserDataEnabled` ⌋*(RS_Nm_02503, RS_Nm_02504)*

[SWS_ANM_00040] ⌈ If NM user data is configured it will be sent always in the NM message. ⌋*(RS_Nm_00150)*

Note: the range (in bytes) that contains the user data in the received NM message is defined by `NmUserDataLength`.

Note: UserData does not include the PNI in case the Partial Networking is active. Received and Transmitted UserData does not overlap with the PNI.

7.6 Nm Transmission

7.6.1 Transmission Scheduling

Note: The periodic transmission mode is used in the "Repeat Message State" and "Normal Operation State".

[SWS_ANM_00044] ⌈ If the Repeat Message State is entered ([SWS_ANM_00070]), the transmission of NM message shall be delayed by `NmMsgCycleOffset` after entering the Repeat Message State. ⌋*(RS_Nm_00044, RS_Nm_00150)*

[SWS_ANM_00046] ⌈ If transmission of NM messages has been started and the NM Message Cycle Timer expires, a NM message transmission shall be initiated. ⌋ *(RS_Nm_00044)*

[SWS_ANM_00047] ⌈ If the NM Message Cycle Timer expires it shall be restarted with `NmMsgCycleTime`. ⌋*(RS_Nm_00044, RS_Nm_00150)*

7.7 Nm Reception

[SWS_ANM_00048] ⌈ The NM shall invoke the user data notification every time a changed user data information has been received in the NM message. ⌋ *(RS_Nm_02504)*

Note: A change in the user data implies that at least 1 Bit within the received user data has changed its value. User data information can be changed by any node within the physical network.

7.8 Partial Networking

7.8.1 Partial Network State Machine

The partial network state machine mentioned in Figure 7.1 is supposed to be implementation specific. Note: Although beeing implementation specific, the implemented behaviour shall conform to the Partial Networking requirements described in [3].

7.8.2 Rx Handling of NM messages

Note : Reception Handling of PN Information as described in [3] is switched on/off by `NmPnEnabled`

[SWS_ANM_00051] ⌈ If `NmPnEnabled` is `TRUE` and the PNI bit in the received NM message is 1, the NM shall invoke the NetworkHandle Notifier, every time a BitVector has been received in the NM message that leads to a change in the actual request state of a relevant PN or a channel. ⌋*(RS_Nm_00150)*

Note: If `NmPnEnabled` is `TRUE` and the PNI bit in the received NM message is 0, NM module still will process the user data information.

[SWS_ANM_00077] ⌈ If `NmPnEnabled` is `TRUE`, the getter for the NetworkHandle shall provide the (aggregated) request state of the PNs instead of mirroring the last received Partial Network information. ⌋*(RS_Nm_00150)*

Note: PnResetTime specified in [3] shall be configured by `NmPnResetTime`. Note: `NmPnResetTime` shall be configured to a value greater than `NmMsgCycleTime`.

7.8.3 Tx Handling of NM messages

Note: `NmPnEnabled` enable/disable the PNI transmission as specified in chapter "Handling of Tx NM messages" of [3].

Note: The usage of the CBV is mandatory in case Partial Networking is used. This has to be ensured by configuration in the respective platform.

7.8.4 NM message Filter Algorithm

[SWS_ANM_00055] ⌈ The range (in bytes) that contains the PN request information (PN Info Range) in the received NM message shall be defined by PN Info Offset (`NmPnInfoOffset`) starting from byte 0 and PN Info Length (`NmPnInfoLength`). ⌋ *(RS_Nm_02520)*

Example:

- PN Info Offset = 3

- PN Info Length = 2

In the above example only Byte 3 and Byte 4 of the NM message contain PN request information.

Note: every bit of the PN Info Range represents one Partial Network. If the bit is set to 1 the Partial Network is requested. If the bit is set to 0 there is no request for this PN.

The Adaptive Application requesting the partial network, shall encode the information in the form of Bit fields and use the methods `RequestNetwork` and `ReleaseNetwork`.

[SWS_ANM_00081] ⌈ The Nm shall optionally filter out messages containing PN request information if they do not contain at least one bit set to 1 that corresponds to a pnc that is configured for this ECU. ⌋*(RS_Nm_00150)* Note: When activated the Nm Message Filter Algorithm will filter out any Nm Message not containing at least one relevant PN beeing requested (its Bit in the PN vector set to 1). Note: (Independent) Message Filter Mask like in Classic Platform is currently not forseen.

8 API specification

The `Network Management` does not contain any APIs. All the functionality is provided via Services.

9 Service Interfaces

The `Network Management` does not provide any services towards applications. Requests for networks shall be received from `State Management` and are defined in Appendix A.

10 Configuration parameters

The following chapter summarizes the configuration parameters referred in this specification.

Parameter	Description
NmTimeoutTime	Network Timeout for NM packets. It denotes the time in [s] how long the NM shall stay in the Network Mode before transition into Prepare Bus-Sleep Mode shall take place.
NmMsgCycleTime	Period of a NM-message. It determines the periodic rate and is the basis for transmit scheduling.
NmRepeatMessageTime	Timeout for Repeat Message State. It defines the time in seconds how long the NM shall stay in the Repeat Message State.
NmWaitBusSleepTime	Timeout for bus calm down phase. It denotes the time in [s] how long the NM shall stay in the Prepare Bus-Sleep Mode before transition into Bus-Sleep Mode shall take place.
NmMsgCycleOffset	Time offset in the periodic transmission node. It determines the start delay of the transmission.
NmPduNidPosition	Defines the position of the source node identifier within the NM message.
NmPduCbvPosition	Defines the position of the control bit vector within the NM message.
NmPnInfoLength	Specifies the length (in bytes) of the PN request information in the NM message.
NmPnInfoOffset	Specifies the offset (in bytes) of the PN request information in the NM message.
NmUserDataLength	Specifies the length (in bytes) of the user data information in the NM message. User data excludes the PN information.
NmPnEnabled	Enables or disables support of partial networking. false: Partial networking Range not supported true: Partial networking supported.
NmPnResetTime	Specifies the runtime of the reset timer in seconds. This reset time is valid for the reset of PN requests by received messages. The value shall be the same for every channel. Thus it is a global config parameter.
NmNodeId	Specifies the Node Identification of this NM node.
NmUserDataEnabled	Enables or disables support of user data information.

Table 10.1: Configuration parameters

Note: The implementer should handle the above-mentioned configuration themselves (eg: as an included header file), as these configuration will not be part of the manifest file in this release of the specification

A Interfaces to other Functional Clusters (informative)

A.1 Overview

AUTOSAR decided not to standardize interfaces which are exclusively used between Functional Clusters (on platform-level only), to allow efficient implementations, which might depend e.g. on the used Operating System.

This chapter provides informative guidelines how the interaction between Functional Clusters looks like, by clustering the relevant requirements of this document to describe Inter-Functional Cluster (IFC) interfaces. In addition, the standardized public interfaces which are accessible by user space applications (see chapters 8 and 9) can also be used for interaction between Functional Clusters.

The goal is to provide a clear understanding of Functional Cluster boundaries and interaction, without specifying syntactical details. This ensures compatibility between documents specifying different Functional Clusters and supports parallel implementation of different Functional Clusters. Details of the interfaces are up to the platform provider. Additional interfaces, parameters and return values can be added.

A.2 Interface Tables

A.2.1 Network Request/Release

Since Applications cannot directly interact with `NM`, the functional clusters does not offer any APIs or Services in chapter 8 and 9. The following service interface with it's elements represents an example implementation on how `State Manager` can interact with `Network Management`.

Port

Name	NetworkHandle_{NmNode}		
Kind	ProvidedPort	***Interface***	`NetworkHandle`
Description	Provides information about network management status.		
Variation	`FOR NmNode : MODEL.filterType("NmNode");`		

Table A.1: Port - NetworkHandle_{NmNode}

Service Interface

Name	NetworkHandle
NameSpace	ara::nm

Table A.2: Service Interfaces - NetworkHandle

Fields

Name	NetworkHandle
Description	Set to true by Network Management when (partial) networks are active or by State Management to activate (partial) networks
Type	`bool`
HasGetter	true
HasNotifier	true
HasSetter	true

Table A.3: Service Interface NetworkHandle - Field: NetworkHandle