Research & Vehicle Technology
"Infotainment Systems Product Development"

Feature – Message Center Client v3

Infotainment Subsystem Part Specific Specification (SPSS)

Version 1.1

UNCONTROLLED COPY IF PRINTED

Version Date: August 23, 2019

FORD CONFIDENTIAL

Revision History

Date	Version	Notes	
April 12, 2019	**1.0**	**Initial Release**	
August 23, 2019	**1.1**	**Updated Release**	
	MCv3-IIR-REQ-347047/B-MessageCenterClient_Tx		hzubert: MCCenterStackHmiNotification_St
	MCv3-MD-REQ-362275/A-MCCenterStackHmiNotification_St		hzubert: initial revision
	STR-635758/B-Functional Definition		hzubert: added MCv3-FUN-REQ-362264
	UCD-REQ-348254/B-Use Case Diagram Message Center System		hzubert: added "Broadcast Center Stack HMI Notification Capability"
	MCv3-FUN-REQ-362264/A-Message Center Broadcast Notification Capability Status		hzubert: initial revision
	MCv3-UC-REQ-362265/A-Broadcast Center Stack HMI Notification Capability		hzubert: initial revision
	MCv3-ACT-REQ-362266/A-Broadcast Center Stack HMI Notification Capability		hzubert: initial revision
	MCv3-SD-REQ-362267/A-Broadcast Center Stack HMI Notification Capability at Startup		hzubert: initial revision
	MCv3-SD-REQ-362268/A-Broadcast Center Stack HMI Notification Capability changes to On		hzubert: initial revision
	MCv3-SD-REQ-362271/A-Broadcast Center Stack HMI Notification Capability changes to Off		hzubert: initial revision

Table of Contents

1 Definitions / Acronyms

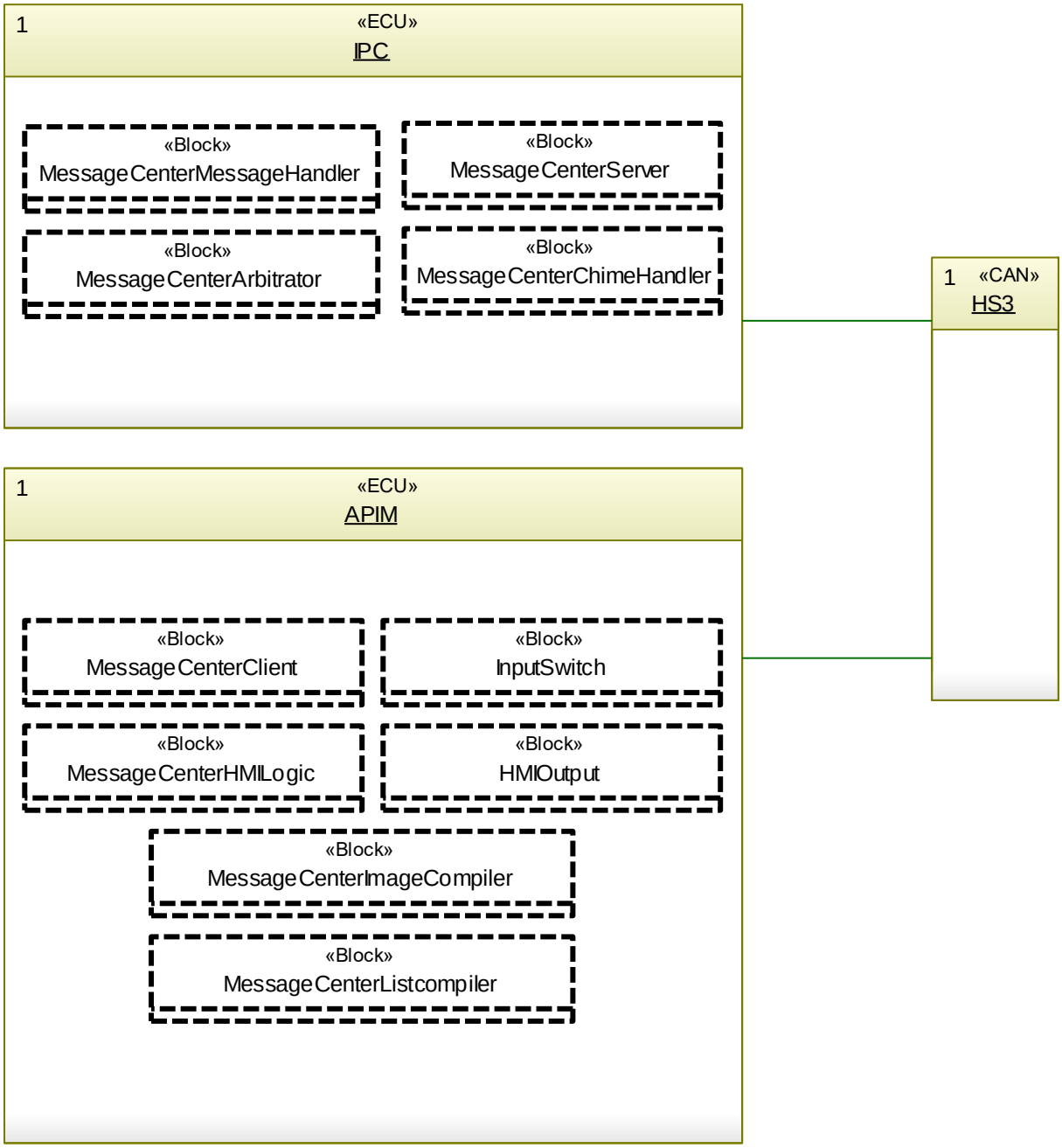
DDx – Dynamic Data with number “x” e.g. 1, 2 or 3
MC – Message Center

2 Architectural Design

2.1 Overview

2.2 Deployment

2.2.1 DEP-REQ-348163/A-Message Center v3 Deployment



2.3 MCv3-IIR-REQ-347047/B-MessageCenterClient_Tx

2.3.1 MCv3-MD-REQ-347049/A-MCPresentation_St

Message Type: Status

This method is used from MessageCenter Client to MessageCenter Server to indicate current display status for an active Message Center notification.

Name	Literals	Value	Description
MsgID	-	-	
	MsgID	0x0 – 0x7FF	The "MsgID" parameter is used to indicate which Message Center notification status is updated. (see lookup table for more information)
MsgStatus	-	-	
	0x0	Invalid	Is used to state out that MsgID & PresentationStatus do not have valid information
	0x1	Valid	Is used to state out that MsgID & PresentationStatus do have valid information
PresentationStatus	-	-	The "PresentationStatus" parameter is used to indicate the current status of a Message Center notification. It is also used to indicate if a chime shall be played or stopped.
	Inactive	0x0	"Inactive" is used if the status is unknown (e.g. at startup while booting).
	ActiveGranted	0x1	"ActiveGranted" (new warning) is used to indicate that an active Message Center notification is presented via car HMI.
	UpdateGranted	0x2	"UpdateGranted" (old warning) is used to indicate that an updated Message Center notification is presented via car HMI.
	Deactivated	0x3	"Deactivated" is used to indicate that a Message Center notification is discarded from the car HMI.
	NotUsed_1	0x4	Reserved for future extensions
	NotUsed_2	0x5	
	NotUsed_3	0x6	
	NotUsed_4	0x7	

2.3.2 MCv3-MD-REQ-347050/A-MCConfirmationSelection_Ind

Message Type: Indication

This method is used from the MessageCenter Client to the MessageCenter Server to indicate which type of confirmation/selection the user made on the currently granted Message Center notification.

Name	Literals	Value	Description
MsgID	-	-	
	MsgID	0x0 – 0x7FF	The "MsgID" parameter is used to indicate which Message Center notification was selected by the user (see lookup table for more information).
MsgStatus	-	-	
	Invalid	0x0	Is used to state out that MsgID & SelectionStatus do not have valid information
	Valid	0x1	Is used to state out that MsgID & SelectionStatus do have valid information
SelectionStatus	-	-	The SelectionStatus is used to indicate if a warning is confirmed via OK input or if a selection has been choosen.
	OK	0x0	“OK” is used to indicate that the user confirmed an OK input.
	Choice1	0x1	“Choice1-7” is used to indicate that the user made a selection on a choice element.
	Choice2	0x2	
	…	…	
	Choice7	0x7	

2.3.3 MCv3-MD-REQ-362275/A-MCCenterStackHmiNotification_St

Message Type: Status

This method is used from MessageCenter Client to MessageCenter Server to reflect actual HMI notification status. This works like a heartbeat and is transmitted periodically.

Name	Literals	Value	Description
Status	-	-	The "Notification Status" parameter is used to indicate the current status of the Message Center Client HMI and if it is capable to present notifications to the user.
	Off	0x0	"Off" is used if the status shows “not capable” to present notifications. This can be while boot up or while some failure cases HMI is not capable to present notifications.
	On	0x1	"On" is used if the status shows “capable” to present notifications via car HMI.

2.4 MCv3-IIR-REQ-347051/A-MessageCenterClient_Rx

2.4.1 MCv3-MD-REQ-347052/A-MCEventNotification_Rq

Message Type: Status

This method is used from MessageCenter Server to inform MessageCenter Client that a Message Center notification update occurred.

Name	Literals	Value	Description
MsgID	-	-	
	ID	0x00 – 0x7FF	The "Message ID" parameter is used to indicate which MessageCenter notification is updated (see lookup table for more information)
MsgStatus	-	-	"MsgStatus" is used to indicate actual status of MessageCenter events.
	Invalid	0x0	“Invalid” is used if data of this message is invalid e.g. if no data is available e.g. at startup or if no MessageCenter notification is active. As well as MsgID, HighlightedChoice and DynamicData1 to 3 do not have valid information.
	Activate	0x1	"Activate" (new warning) is used if a MessageCenter notification is set to active (-> allocated).
	Update	0x2	“Update” (old warning) is used if a MessageCenter notification, which was “Active” before, shall be shown again. Including data update.
	Deactivate	0x3	"Deactivate" is used if an active MessageCenter notification is set to inactive (->deallocated).
HighlightedChoice	-	-	“Highlighted Choice” is used to indicate the highlighted choice
	NoChoice	0x0	“NoChoice” is used if no choice element is highlighted
	Choice1	0x1	“Choice 1-7” is used if one of choice 1 to 7 is highlighted.
	Choice2	0x2	
	…	…	
	Choice7	0x7	
DynamicData1	-	-	Dynamic Data 1 with up to 2 Bytes
	RAW	2 Byte	
DynamicData2	-	-	Dynamic Data 2 with up to 2 Bytes
	RAW	2 Byte	
DynamicData3	-	-	Dynamic Data 3 with up to 2 Bytes
	RAW	2 Byte	

3 General Requirements

3.1 MCv3-REQ-347870/A-Language setting

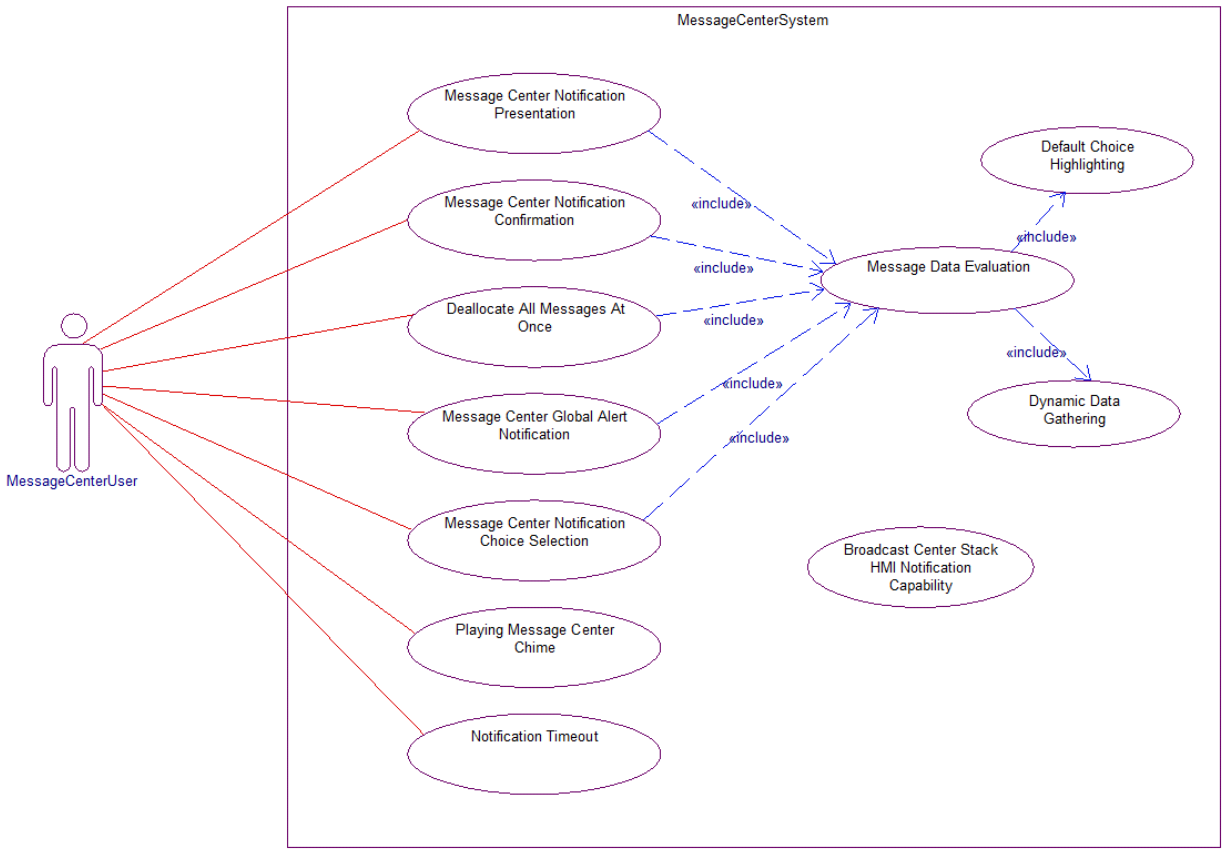
Notifications have to be in set language all the time.

3.2 MCv3-REQ-348225/A-Duplicate bus signals and messages

If there is more than one signal/message on bus system, with same values, following the other, they have to be ignored.

4 Functional Definition

4.1 UCD-REQ-348254/B-Use Case Diagram Message Center System



4.2 MCv3-FUN-REQ-347041/A-Message Center Notification Presentation

4.2.1 Requirements

4.2.2 Use Cases

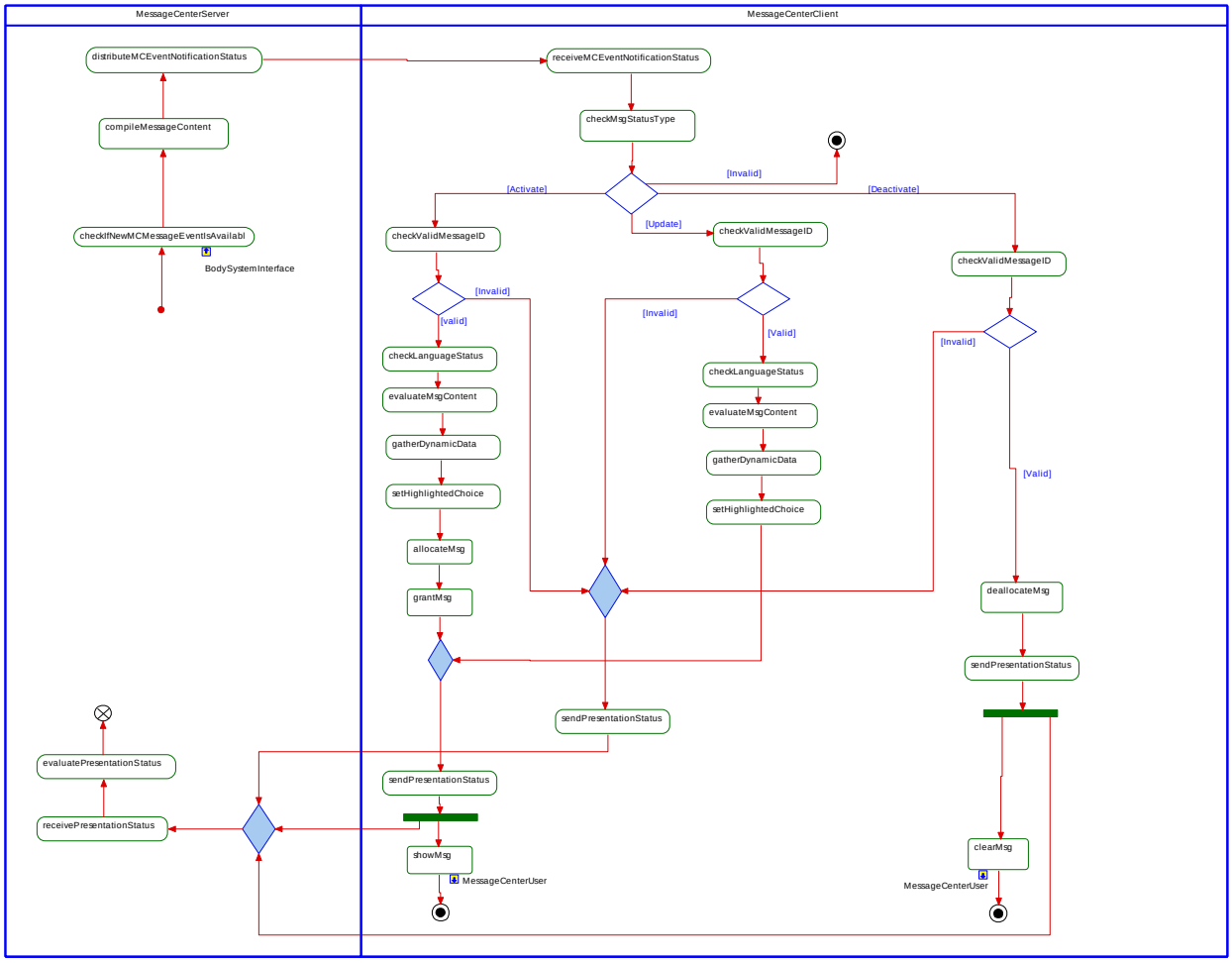
4.2.2.1 MCv3-UC-REQ-347043/A-Message Center Notification Presentation

Actors	Message Center User Vehicle System
Pre-conditions	A trigger for a Message Center notification occurred
Scenario Description	MessageCenter Server compiles related message content and transmits this notification message to MessageCenter Client. This checks validity (correct message type and ID), checks in lookup table if any choice fields or similar is available, actual set language status, evaluates content of received bus message, reads and acquires type of all dynamic data raw fields, interprets these accordingly, reads if to highlight any choice element, allocates and displays or deactivates this notification and sends according status back to MessageCenter Server
Post-conditions	Message Center notification is created (activated) and displayed, presented to user again (updated) or removed (deactivated).
List of Exception Use Cases	NA
Interfaces	G-HMI MCEventNotification_Rq MCPresentation_St

4.2.3 Activity Diagrams

4.2.3.1 MCv3-ACT-REQ-347044/A-Message Center Notification Presentation

Activity Diagram



4.2.4 Sequence Diagrams

4.2.4.1 MCv3-SD-REQ-347045/A-Message Center Notification Presentation_StatusType is Activate, No Message Granted

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of MsgID 15 and transmits this notification message to MessageCenter Client. This checks validity (correct message type and ID), checks in lookup table that no choice field or similar is available, checks actual set language status, evaluates content of received bus message, reads and acquires type of all dynamic data raw fields and sees no data to be used, reads to highlight no choice element, allocates, grants and displays this notification and sends active granted status back to MessageCenter Server.

MessageCenter Server receives this notification status and evaluates it to proceed accordingly.

Constraints

Pre-condition

System is on.

Pre-condition

Trigger for a new notification occurred.

Pre-condition

No Message granted (message stack is empty).

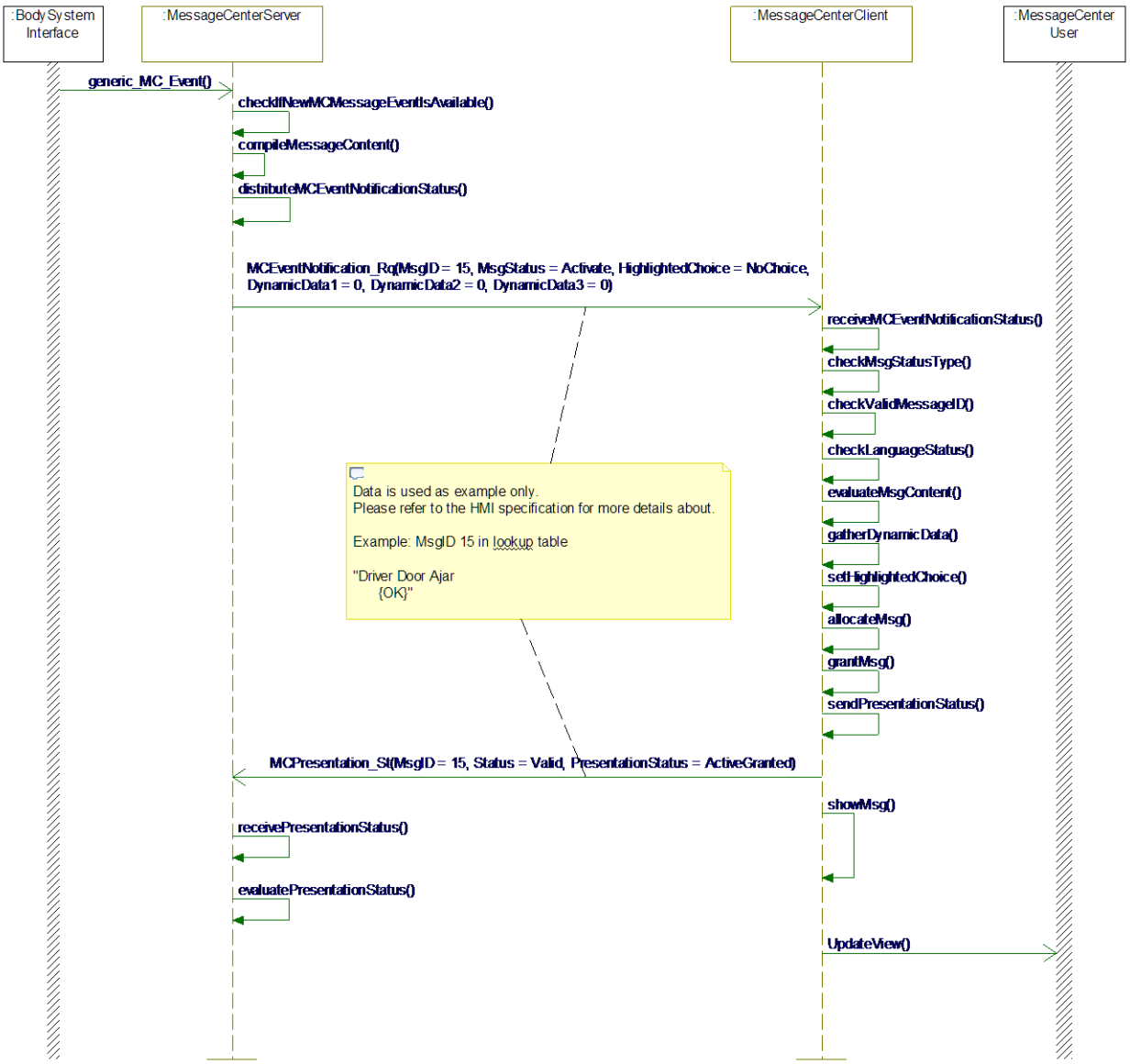
Post-condition

New MC notification is presented to the user.

Post-condition

An “active” presentation state is sent back to MessageCenter Server.

Sequence Diagram



4.2.4.2 MCv3-SD-REQ-347372/A-Message Center Notification Presentation_StatusType is Deactivate, Received Event is Granted, Stack is Empty

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of MsgID 15 and transmits this notification message to MessageCenter Client. This checks validity (correct message type and ID), deallocates this notification (removing it from screen and stack) and sends deactivated status back to MessageCenter Server.

MessageCenter Server receives this notification status and evaluates it to proceed accordingly.

Constraints

Pre-condition

System is on.

Pre-condition

Trigger for this message ID is not available any more.

Pre-condition

The MC notification with this message ID is granted.

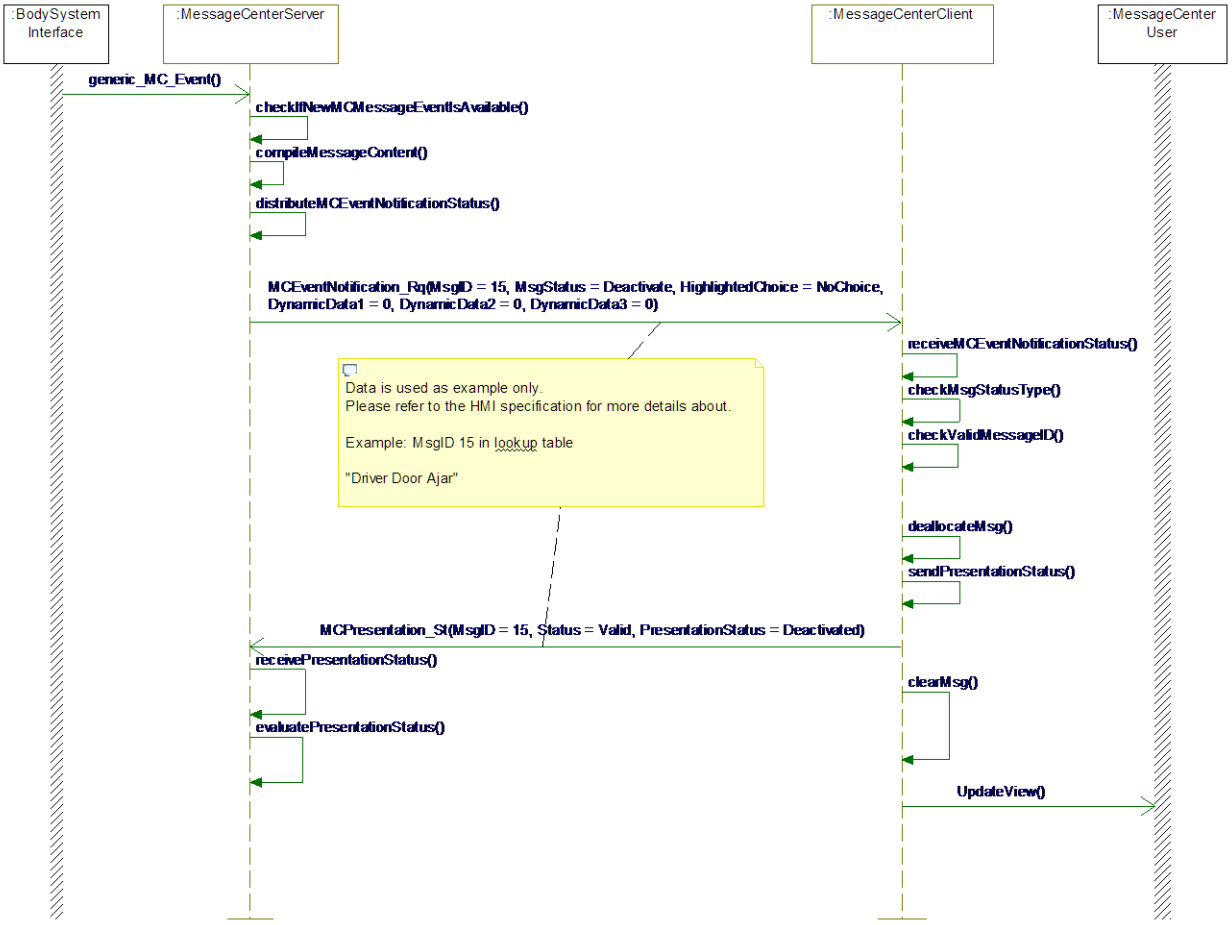
Post-condition

This MC notification has been removed from screen and stack.

Post-condition

An actual presentation state is sent back to MessageCenter Server.

Sequence Diagram



4.2.4.3 MCv3-SD-REQ-347373/A-Message Center Notification Presentation_StatusType is Activate, Choice 1 highlighted, No Message Granted

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of MsgID 124 and transmits this notification message to MessageCenter Client. This checks validity (correct message type and ID), checks in lookup table that two choice fields are available, checks actual set language status, evaluates content of received bus message, reads and acquires type of all dynamic data raw fields and sees no data to be used, reads to highlight first choice element, allocates, grants and displays this notification and sends active granted status back to MessageCenter Server.

MessageCenter Server receives this notification status and evaluates it to proceed accordingly.

Constraints

Pre-condition

System is on.

Pre-condition

Trigger for a new notification occurred.

Pre-condition

No Message granted (message stack is empty).

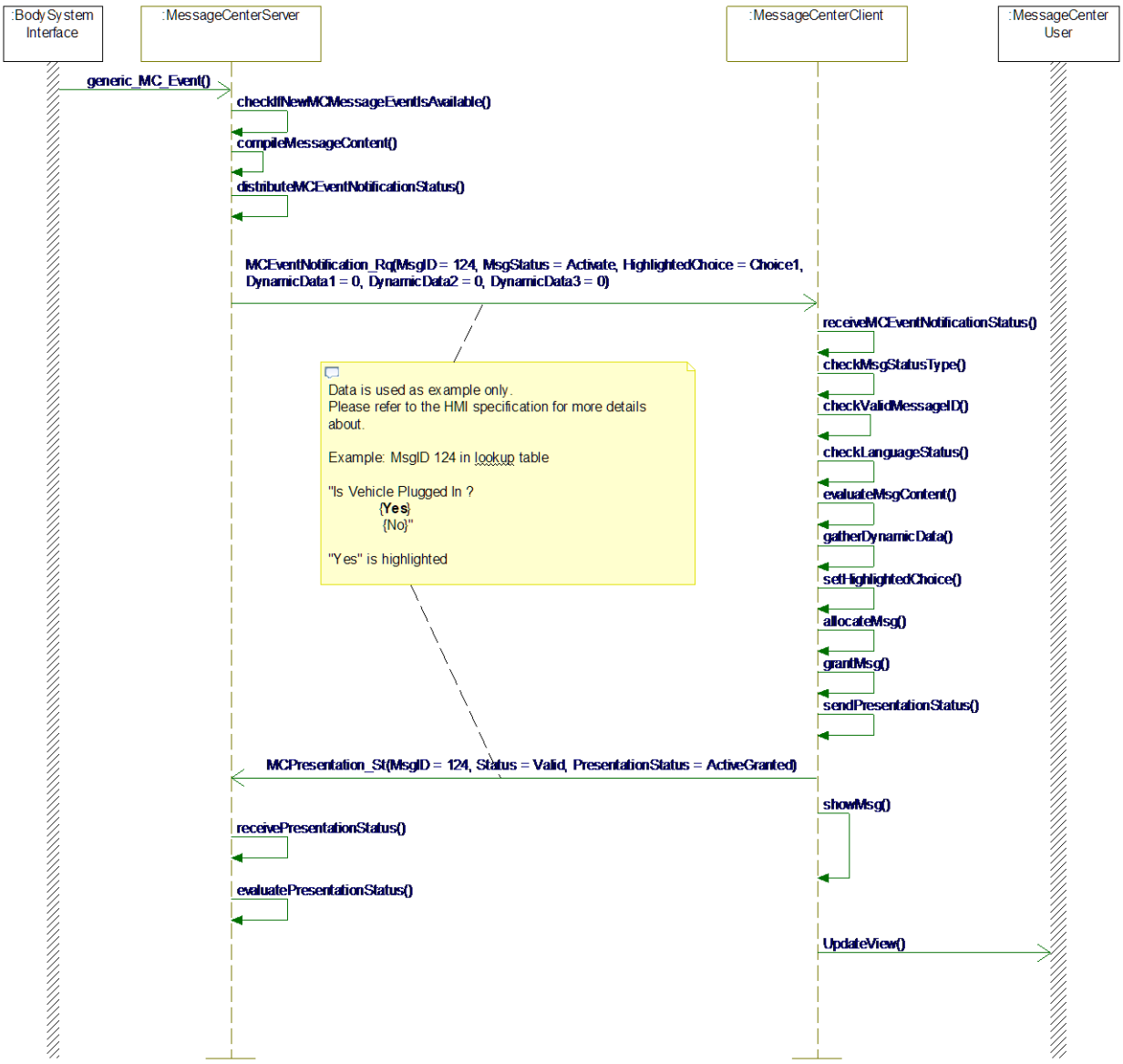
Post-condition

New MC notification is presented to the user.

Post-condition

An “active” presentation state is sent back to MessageCenter Server.

Sequence Diagram



4.2.4.4 MCv3-SD-REQ-347374/A-Message Center Notification Presentation_StatusType is Activate, No Message Granted_LocalDynamicData

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of MsgID 211 and transmits this notification message to MessageCenter Client. This checks validity (correct message type and ID), checks in lookup table that no choice field or similar is available, checks actual set language status, evaluates content of received bus message, reads and acquires type of all dynamic data raw fields and sees, data to be shown, is locally taken from, reads to highlight no choice element, allocates, grants and displays this notification and sends active granted status back to MessageCenter Server.

MessageCenter Server receives this notification status and evaluates it to proceed accordingly.

Constraints

Pre-condition

System is on.

Pre-condition

Trigger for a new notification occurred.

Pre-condition

No Message granted (message stack is empty).

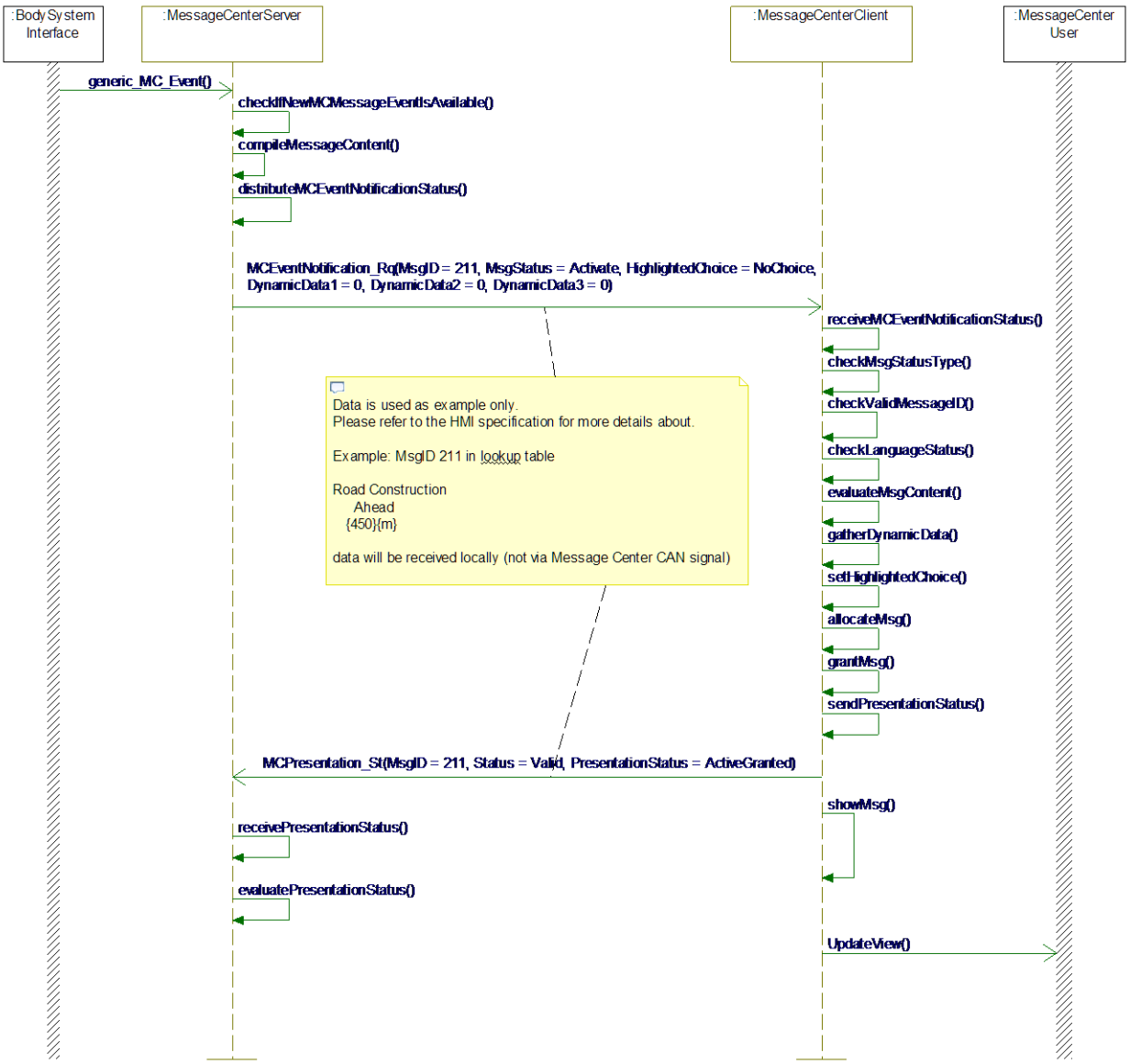
Post-condition

New MC notification is presented to the user.

Post-condition

An “active” presentation state is sent back to MessageCenter Server.

Sequence Diagram



4.2.4.5 MCv3-SD-REQ-347445/A-Message Center Notification Presentation_StatusType is Activate, No Message Granted_2DynamicDataAsOneInteger

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of MsgID 158 and transmits this notification message to MessageCenter Client. This checks validity (correct message type and ID), checks in lookup table that no choice field or similar is available, checks actual set language status, evaluates content of received bus message, reads and acquires type of all dynamic data raw fields.
In this case, due to entry in lookup table, both (MessageCenter Server and Client) do know to interpret DD1 (=DynamicData1–field) and DD2 as one integer and DD3 as an enumeration literal to present 7 digits (-> see MCv3-FUN-REQ-347021-Dynamic Data Gathering for more details). Then it reads to highlight no choice element, allocates, grants and displays this notification and sends active granted status back to MessageCenter Server.

MessageCenter Server receives this notification status and evaluates it to proceed accordingly.

Example (please refer to the HMI specification for more details about data):
MsgID 158 in lookup table with DD3 for 7 digits to show.

Factory Keypad
Code
{ 4 3 7 5 9 2 4 }

use DD1 and DD2 as one integer:

DD1 = 0x42, DD2 = 0xC574 => value is 0x42C574

how to calculate manually to verify:
DD2 = 0xC574 -> 0xC574 -> 50548
DD1 = 0x42 -> 0x420000 -> 4325376

4375924

Constraints

Pre-condition

System is on.

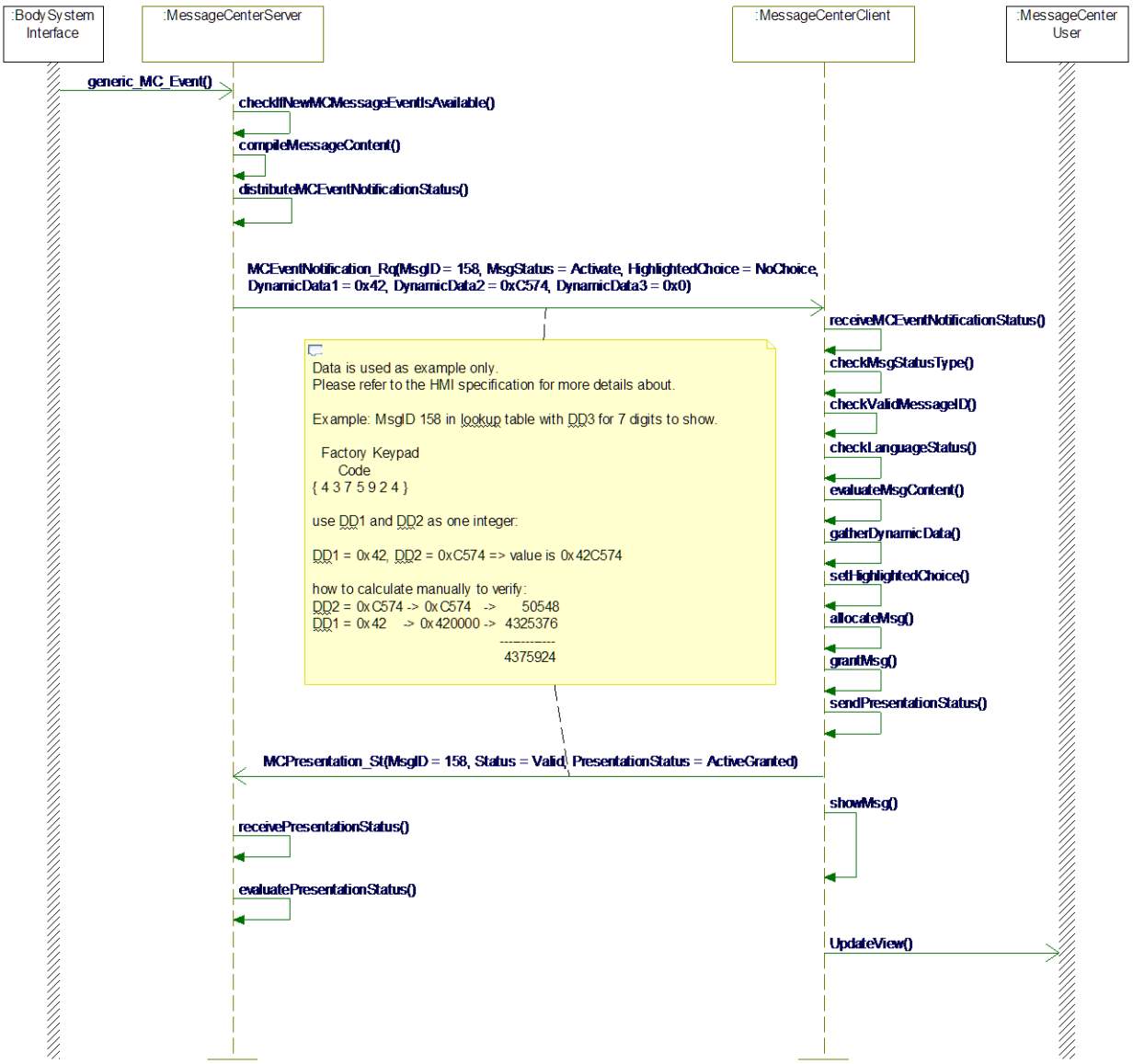
Pre-condition

No message granted.

Post-condition

New MC notification is presented to the user.

Sequence Diagram



4.2.4.6 MCv3-SD-REQ-348253/A-Message Center Notification Presentation_StatusType is Activate, No Message Granted_2DynamicDataAsOneInteger_leading zeros

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of MsgID 158 and transmits this notification message to MessageCenter Client. This checks validity (correct message type and ID), checks in lookup table that no choice field or similar is available, checks actual set language status, evaluates content of received bus message, reads and acquires type of all dynamic data raw fields.
In this case, due to entry in lookup table, both (MessageCenter Server and Client) do know to interpret DD1 (=DynamicData1–field) and DD2 as one integer and DD3 as an enumeration literal to present 5 digits (-> see MCv3-FUN-REQ-347021-Dynamic Data Gathering for more details). Then it reads to highlight no choice element, allocates, grants and displays this notification and sends active granted status back to MessageCenter Server.

MessageCenter Server receives this notification status and evaluates it to proceed accordingly.

Example (please refer to the HMI specification for more details about data):
MsgID 158 in lookup table with DD3 for 5 digits to show.

Factory Keypad
Code
{ 0 0 1 2 3 }

use DD1 and DD2 as one integer:

DD1 = 0x0, DD2 = 0x7B => value is 0x7B
how to calculate manually to verify:
DD2 = 0x7B -> 0x007B -> 00123
DD1 = 0x0 -> 0x0000 -> 00000

0000000123 -> 5 digits -> "00123"

Constraints

Pre-condition

System is on.

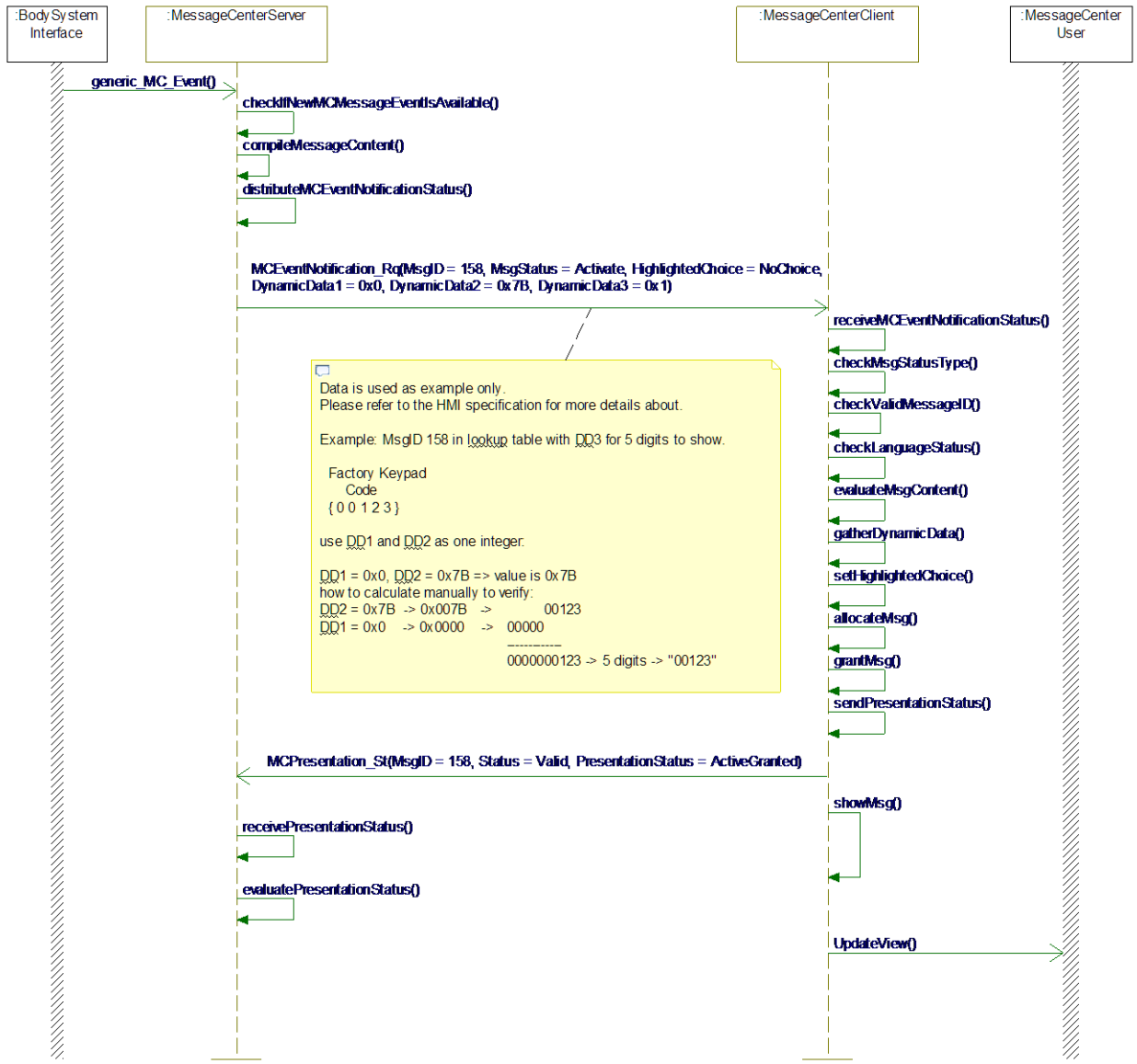
Pre-condition

No message granted.

Post-condition

New MC notification is presented to the user.

Sequence Diagram



4.2.4.7 MCv3-SD-REQ-347446/A-Message Center Notification Presentation_StatusType is Activate, MsgID is not available

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of an unknown MsgID (here e.g. 444) with MsgStatus of activate, update or deactivate and transmits this notification message to MessageCenter Client.

MessageCenter Client checks if message type and message ID are valid. In this case, it can't find this message ID in lookup table and only sends inactive presentation status back to indicate an error.

Constraints

Pre-condition

System is on.

Pre-condition

MsgID is not available in lookup table.

Post-condition

MessageCenter Client sends back an „Inactive" presentation status to MessageCenter Server.

Sequence Diagram

4.2.4.8 MCv3-SD-REQ-347945/A-Message Center Notification Presentation_StatusType is Activate, Received Event is already Granted

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of MsgID 15 and transmits this notification message to MessageCenter Client. This checks validity (correct message type and ID) and sees an activation on an already granted notification.

MessageCenter Client is acting like on a new message (-> see MCv3-SD-REQ-347045-Message Center Notification Presentation_StatusType is Activate, No Message Granted) and sends active granted status back to MessageCenter Server.

MessageCenter Server receives this notification status and evaluates it to proceed accordingly.

Note: This is a robustness reaction and is not part of normal behavior.

Constraints

Pre-condition

System is on.

Pre-condition

Event is already granted or „updated“ the last time.

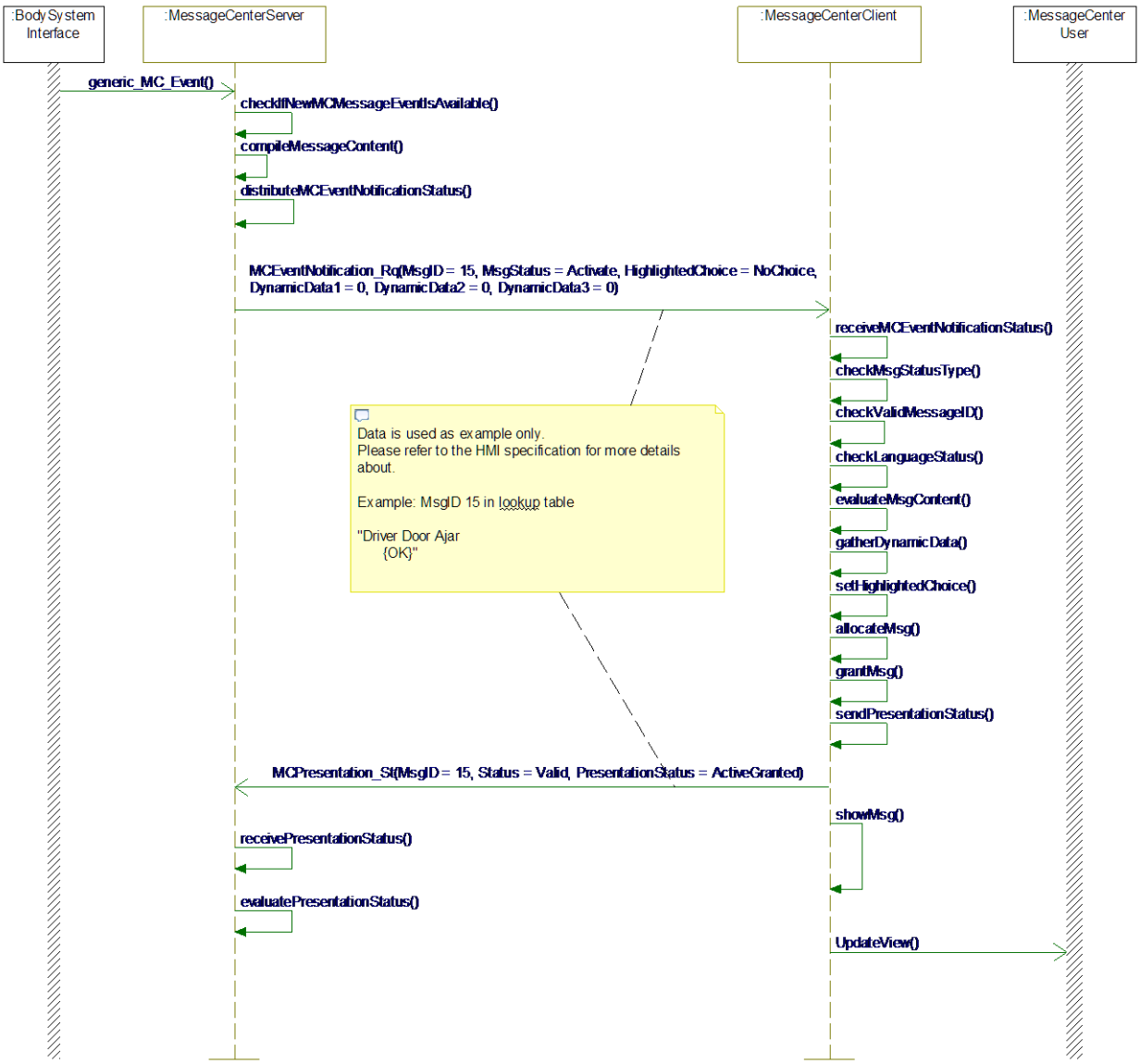
Post-condition

New MC notification is presented to the user.

Post-condition

An “active” presentation state is sent back to MessageCenter Server.

Sequence Diagram



4.2.4.9 MCv3-SD-REQ-347946/A-Message Center Notification Presentation_StatusType is Update, Received Event is not Granted

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of MsgID 15 and transmits this notification message to MessageCenter Client. This checks validity (correct message type and ID) and sees an update on a not granted notification.

MessageCenter Client is acting like it would receive „activate“ on this new message (-> see MCv3-SD-REQ-347045-Message Center Notification Presentation_StatusType is Activate, No Message Granted) and sends active granted status back to MessageCenter Server.

MessageCenter Server receives this notification status and evaluates it to proceed accordingly.

Note: This is a robustness reaction and is not part of normal behavior.

Constraints

Pre-condition

System is on.

Pre-condition

Received event is not granted.

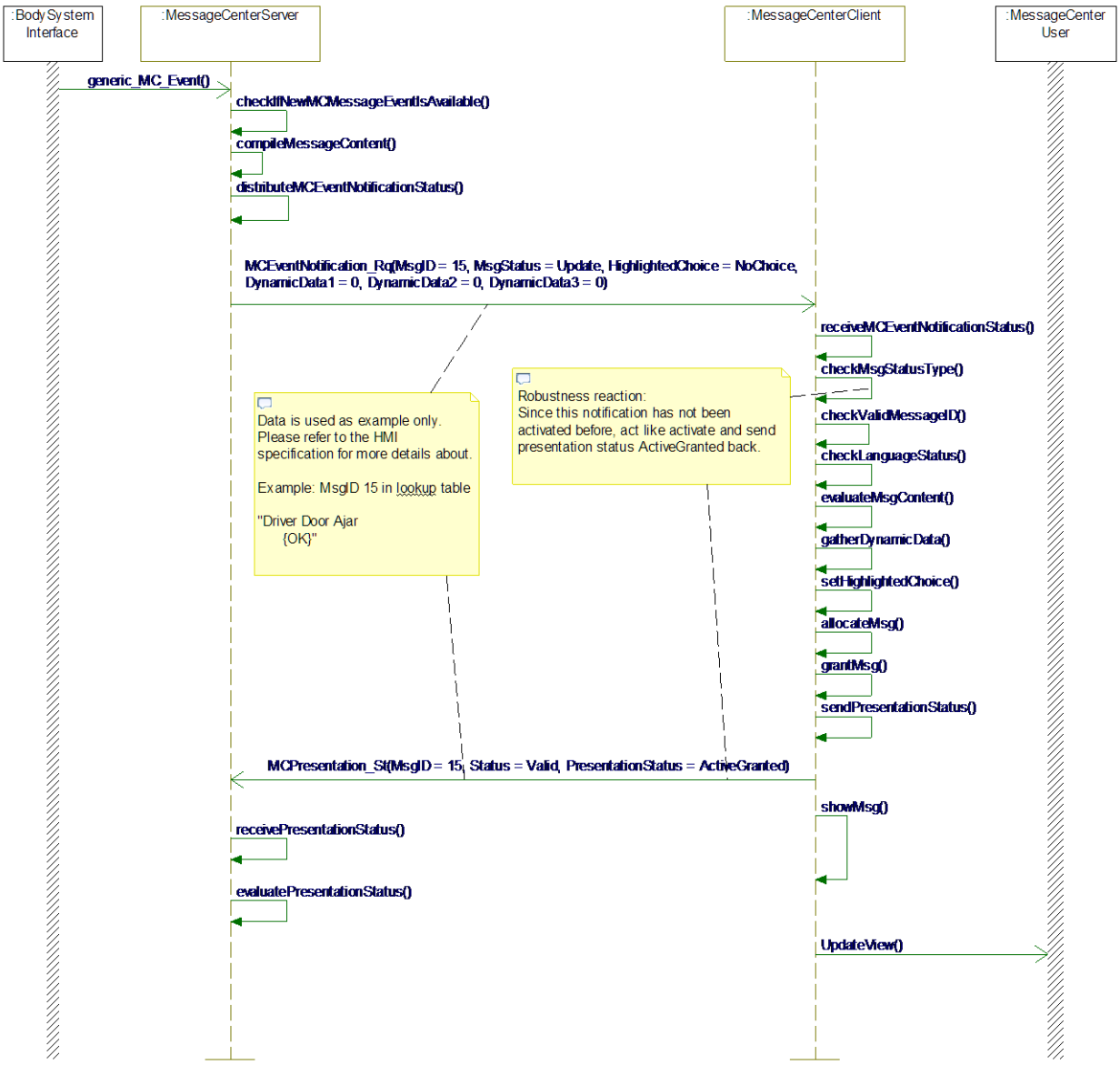
Post-condition

An “old” MC notification is presented to the user.

Post-condition

An “active” presentation state is sent back to MessageCenter Server.

Sequence Diagram



4.2.4.10 MCv3-SD-REQ-347947/A-Message Center Notification Presentation 4 messages in longer sequence showing different MsgStates

Scenarios

Normal Usage

In this example, a sequence of several activations and updates of notifications is shown for better understanding of the message center notification system.

Following scenarios are illustrated:
SC1 activate
SC2 activate
SC3 activate
SC1 update
SC4 activate
SC2 update
SC3 update
SC1 update
SC4 update
SC2 update
SC3 update

Constraints

Pre-condition

System is on.

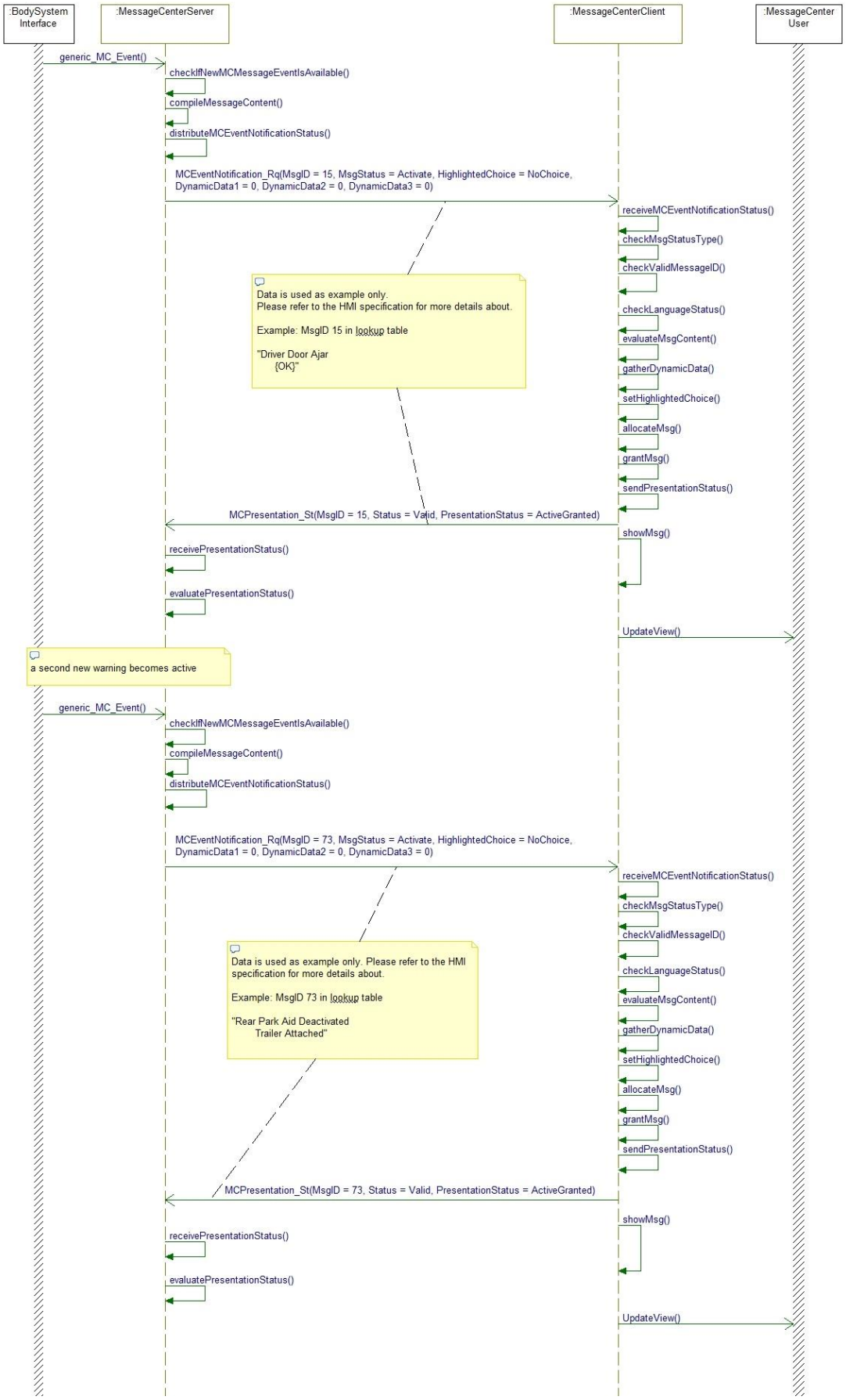
Pre-condition

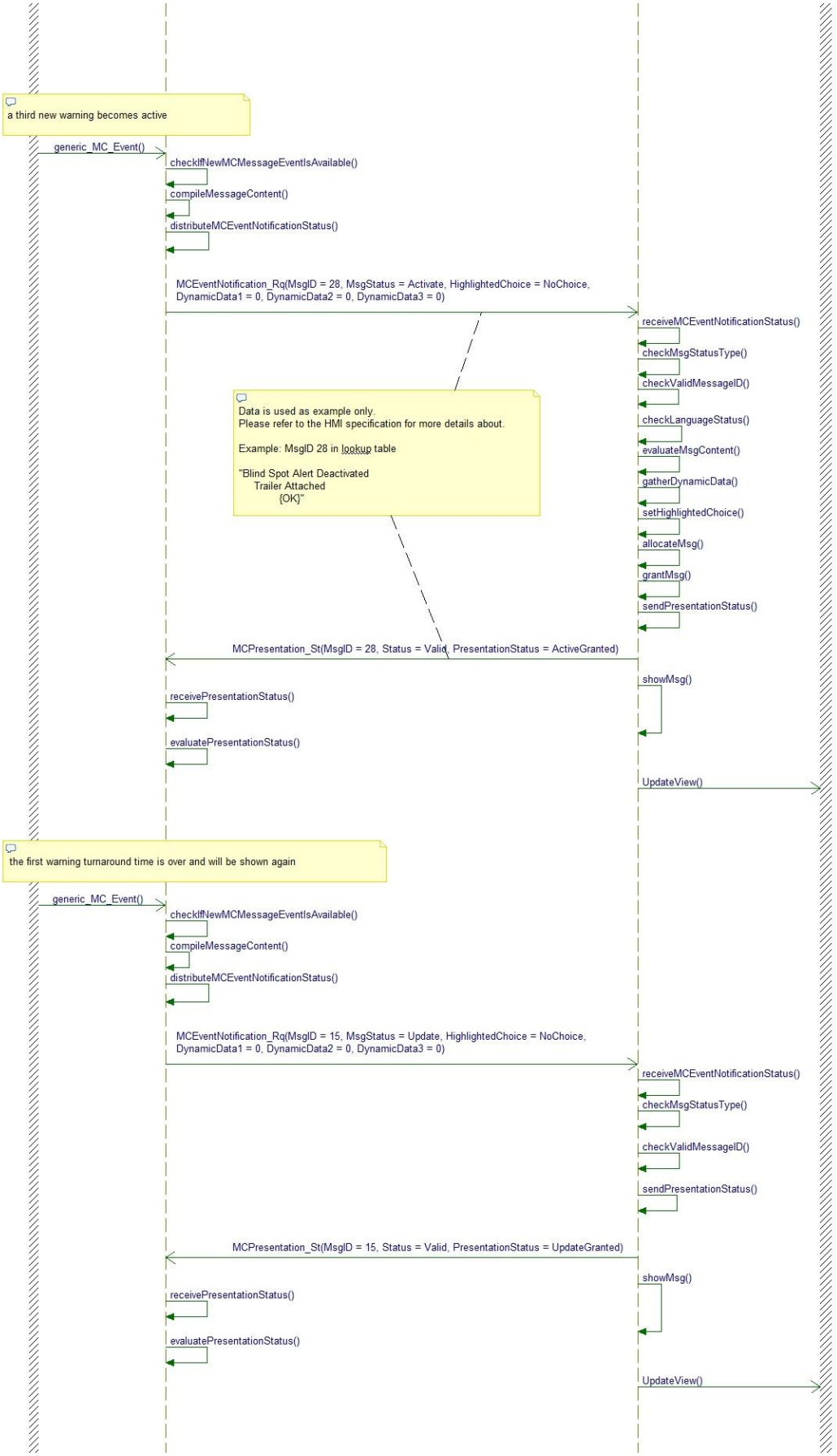
No message granted

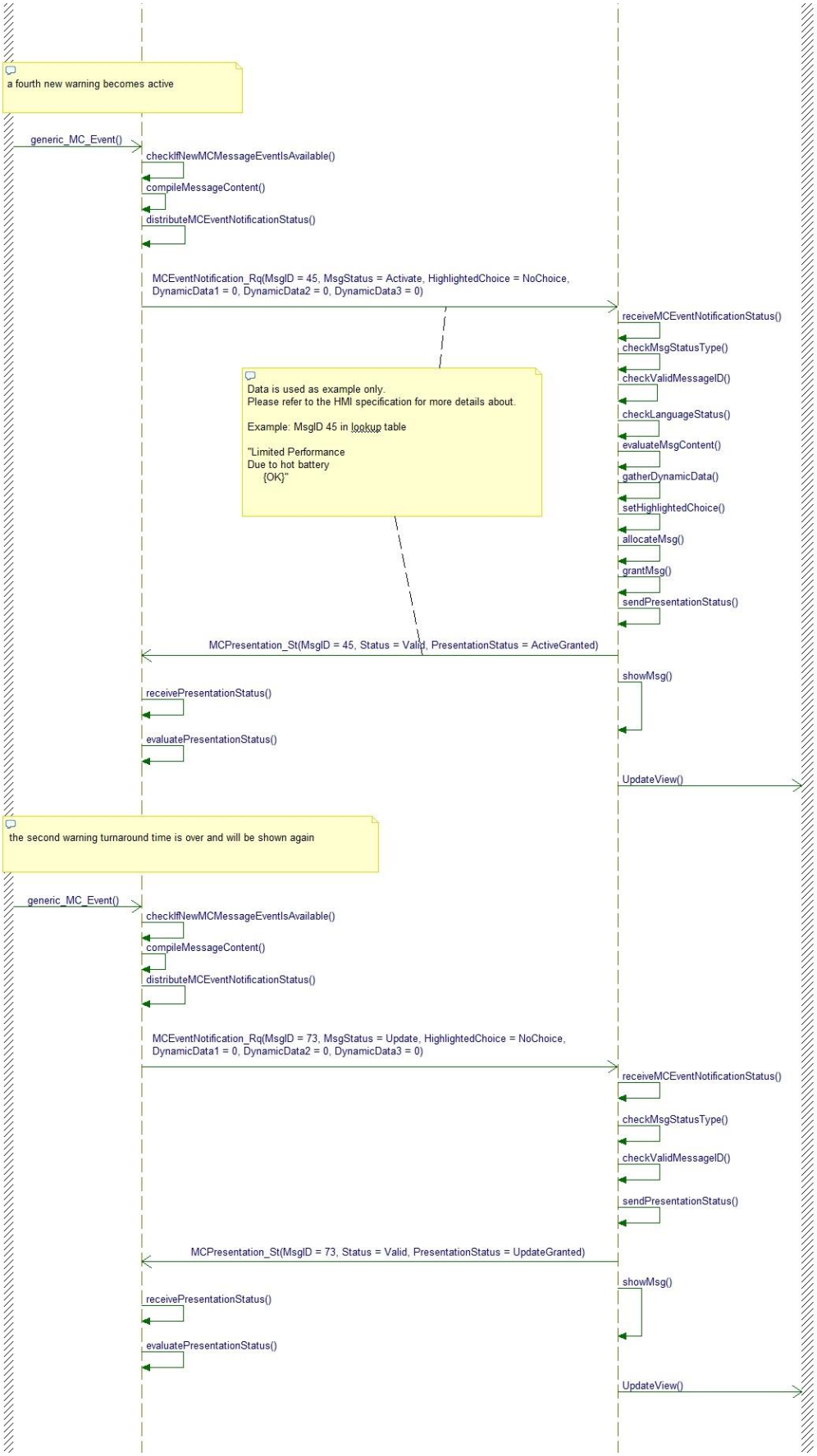
Post-condition

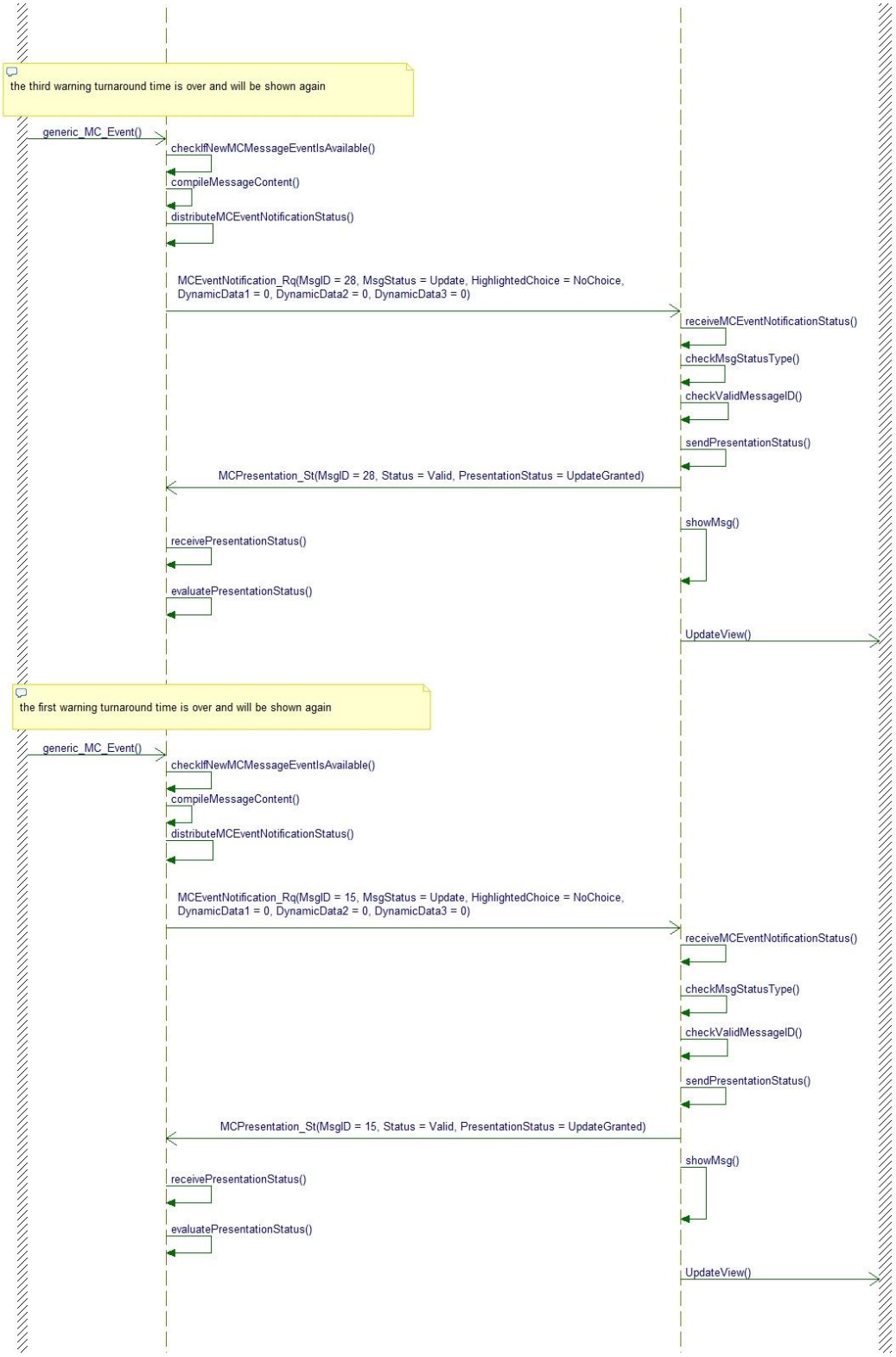
SC3 is presented to the user and SC1, SC4 and SC2 are in queue (stack)

Sequence Diagram

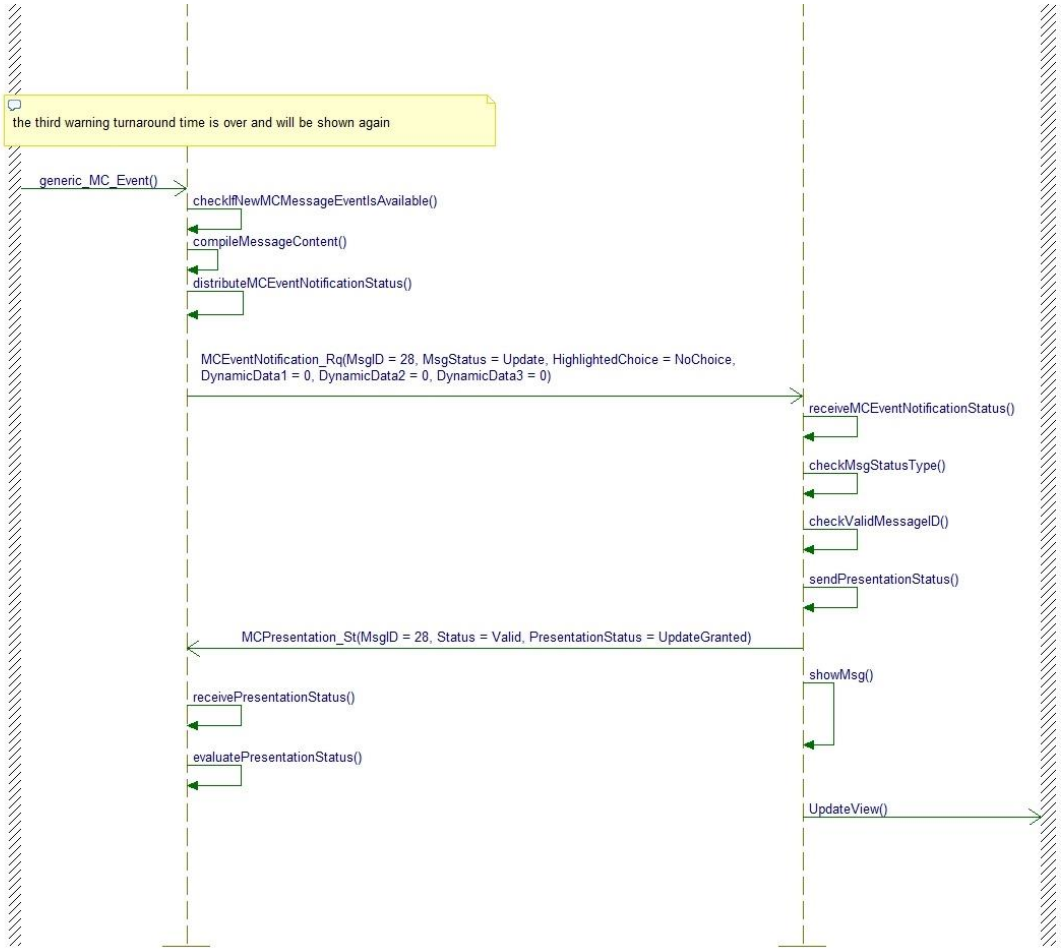

:BodySystem Interface
:MessageCenterServer
:MessageCenterClient
:MessageCenter User
generic_MC_Event()
checkIfNewMCMessageEventIsAvailable()
compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 15, MsgStatus = Activate, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
Data is used as example only.
Please refer to the HMI specification for more details about.
Example: MsgID 15 in lookup table
"Driver Door Ajar
{OK}"
checkLanguageStatus()
evaluateMsgContent()
gatherDynamicData()
setHighlightedChoice()
allocateMsg()
grantMsg()
sendPresentationStatus()
MCPresentation_St(MsgID = 15, Status = Valid, PresentationStatus = ActiveGranted)
showMsg()
receivePresentationStatus()
evaluatePresentationStatus()
UpdateView()
a second new warning becomes active
generic_MC_Event()
checkIfNewMCMessageEventIsAvailable()
compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 73, MsgStatus = Activate, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
Data is used as example only. Please refer to the HMI specification for more details about.
Example: MsgID 73 in lookup table
"Rear Park Aid Deactivated
Trailer Attached"
checkLanguageStatus()
evaluateMsgContent()
gatherDynamicData()
setHighlightedChoice()
allocateMsg()
grantMsg()
sendPresentationStatus()
MCPresentation_St(MsgID = 73, Status = Valid, PresentationStatus = ActiveGranted)
showMsg()
receivePresentationStatus()
evaluatePresentationStatus()
UpdateView()


a third new warning becomes active
generic_MC_Event()
checkIfNewMCMessageEventIsAvailable()
compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 28, MsgStatus = Activate, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
checkLanguageStatus()
evaluateMsgContent()
gatherDynamicData()
setHighlightedChoice()
allocateMsg()
grantMsg()
sendPresentationStatus()
Data is used as example only.
Please refer to the HMI specification for more details about.
Example: MsgID 28 in lookup table
"Blind Spot Alert Deactivated
Trailer Attached
{OK}"
MCPresentation_St(MsgID = 28, Status = Valid, PresentationStatus = ActiveGranted)
showMsg()
receivePresentationStatus()
evaluatePresentationStatus()
UpdateView()
the first warning turnaround time is over and will be shown again
generic_MC_Event()
checkIfNewMCMessageEventIsAvailable()
compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 15, MsgStatus = Update, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
sendPresentationStatus()
MCPresentation_St(MsgID = 15, Status = Valid, PresentationStatus = UpdateGranted)
showMsg()
receivePresentationStatus()
evaluatePresentationStatus()
UpdateView()


a fourth new warning becomes active
generic_MC_Event()
checkIfNewMCMessageEventIsAvailable()
compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 45, MsgStatus = Activate, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
checkLanguageStatus()
evaluateMsgContent()
gatherDynamicData()
setHighlightedChoice()
allocateMsg()
grantMsg()
sendPresentationStatus()
Data is used as example only.
Please refer to the HMI specification for more details about.
Example: MsgID 45 in lookup table
"Limited Performance
Due to hot battery
{OK}"
MCPresentation_St(MsgID = 45, Status = Valid, PresentationStatus = ActiveGranted)
showMsg()
receivePresentationStatus()
evaluatePresentationStatus()
UpdateView()
the second warning turnaround time is over and will be shown again
generic_MC_Event()
checkIfNewMCMessageEventIsAvailable()
compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 73, MsgStatus = Update, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
sendPresentationStatus()
MCPresentation_St(MsgID = 73, Status = Valid, PresentationStatus = UpdateGranted)
showMsg()
receivePresentationStatus()
evaluatePresentationStatus()
UpdateView()


the third warning turnaround time is over and will be shown again
generic_MC_Event()
checkIfNewMCMessageEventIsAvailable()
compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 28, MsgStatus = Update, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
sendPresentationStatus()
MCPresentation_St(MsgID = 28, Status = Valid, PresentationStatus = UpdateGranted)
showMsg()
receivePresentationStatus()
evaluatePresentationStatus()
UpdateView()
the first warning turnaround time is over and will be shown again
generic_MC_Event()
checkIfNewMCMessageEventIsAvailable()
compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 15, MsgStatus = Update, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
sendPresentationStatus()
MCPresentation_St(MsgID = 15, Status = Valid, PresentationStatus = UpdateGranted)
showMsg()
receivePresentationStatus()
evaluatePresentationStatus()
UpdateView()


the fourth warning turnaround time is over and will be shown again
generic_MC_Event()
checkIfNewMCMessageEventIsAvailable()
compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 45, MsgStatus = Update, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
sendPresentationStatus()
MCPresentation_St(MsgID = 45, Status = Valid, PresentationStatus = UpdateGranted)
showMsg()
receivePresentationStatus()
evaluatePresentationStatus()
UpdateView()
the second warning turnaround time is over and will be shown again
generic_MC_Event()
checkIfNewMCMessageEventIsAvailable()
compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 73, MsgStatus = Update, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
sendPresentationStatus()
MCPresentation_St(MsgID = 73, Status = Valid, PresentationStatus = UpdateGranted)
showMsg()
receivePresentationStatus()
evaluatePresentationStatus()
UpdateView()


the third warning turnaround time is over and will be shown again
generic_MC_Event()
checkIfNewMCMessageEventIsAvailable()
compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 28, MsgStatus = Update, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
sendPresentationStatus()
MCPresentation_St(MsgID = 28, Status = Valid, PresentationStatus = UpdateGranted)
showMsg()
receivePresentationStatus()
evaluatePresentationStatus()
UpdateView()

4.2.4.11 MCv3-SD-REQ-349024/A-Message Center Notification Presentation_Error occurred while RAW Data Decoding

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of MsgID 530 and transmits this notification message to MessageCenter Client. This checks validity (correct message type and ID), checks in lookup table that two choice fields are available, checks actual set language status, evaluates content of received bus message, reads and acquires type of all dynamic data raw fields.
In this case, due to entry in lookup table, both (MessageCenter Server and Client) do know to interpret DD1 (=Dynamic Data 1 – field) as an enumeration carrying drive mode. However, the value in transferred DD1 is outside of literal range of this enumeration and cannot be assigned.

Due to that, MessageCenter Client shall just send back inactive presentation state to MessageCenter Server to indicate an error occurred.

MessageCenter Server receives this notification status and evaluates it to proceed accordingly.

Constraints

Pre-condition

System is on.

Pre-condition

MsgStatus can be Activate or Update

Post-condition

An “inactive” presentation state is sent back to MessageCenter Server.

Sequence Diagram

4.3 MCv3-FUN-REQ-347036/A-Message Center Notification Confirmation

4.3.1 Requirements

4.3.2 Use Cases

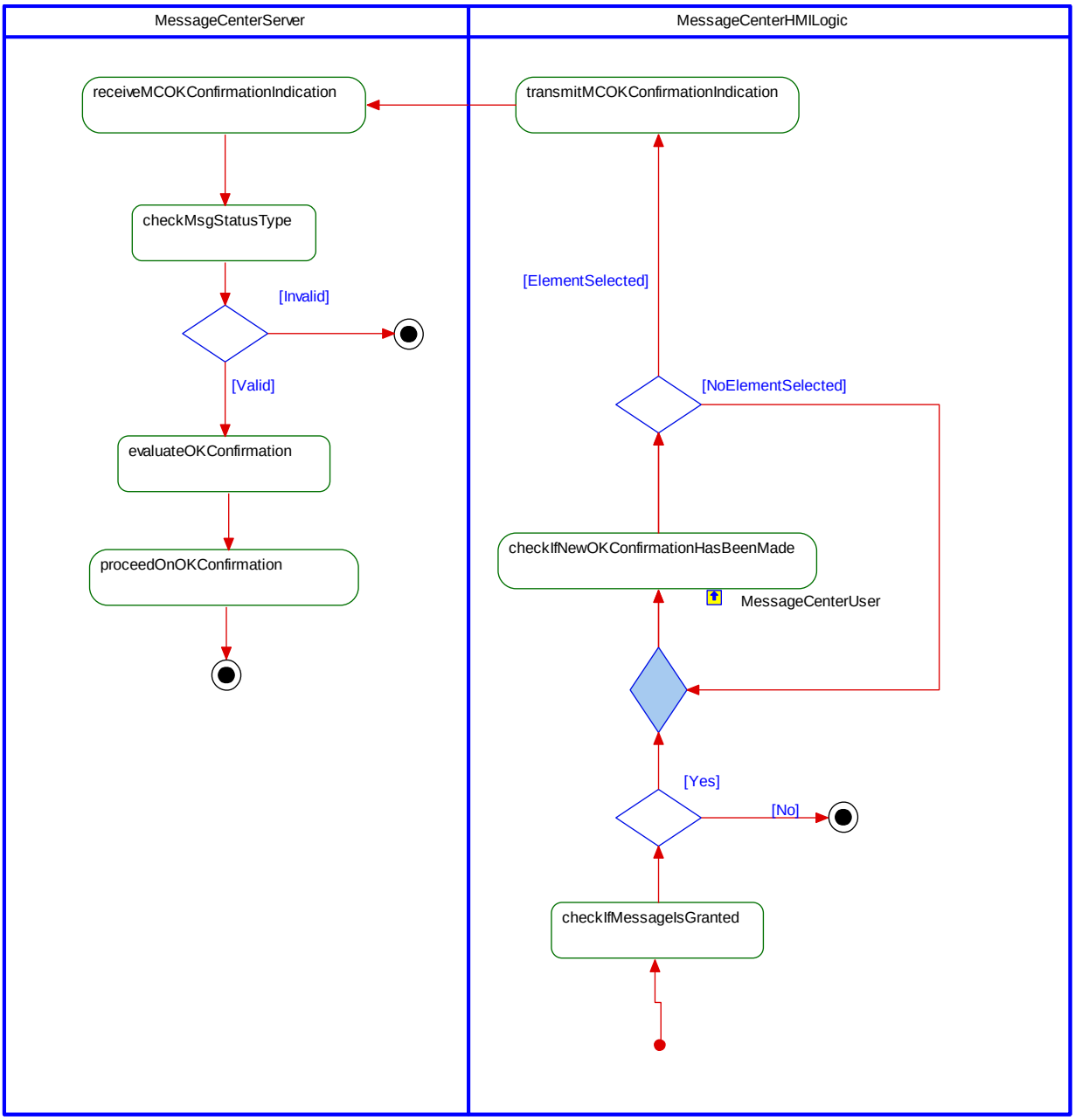
4.3.2.1 MCv3-UC-REQ-347038/A-Message Center Notification Confirmation

Actors	Message Center User
Pre-conditions	A message center notification with a confirmation element (e.g. OK button) is presented to the user (granted).
Scenario Description	User presses confirmation element (e.g. OK button)
Post-conditions	MessageCenter Client sends a confirmation indication information to MessageCenter Server
List of Exception Use Cases	NA
Interfaces	G-HMI MCConfirmationSelection_Ind

4.3.3 Activity Diagrams

4.3.3.1 MCv3-ACT-REQ-347039/A-Message Center Notification Confirmation

Activity Diagram



4.3.4 Sequence Diagrams

4.3.4.1 MCv3-SD-REQ-347375/A-Message Center Notification Confirmation_Messages available, User Choice OK Confirmation, Successor Msg not available

Scenarios

Normal Usage

User presses OK. HMI shows reaction on OK press (e.g. button is highlighted). This confirmation indication is sent to MessageCenter Server.

Constraints

Pre-condition

System is on.

Pre-condition

MC notification with OK button is presented to the user.

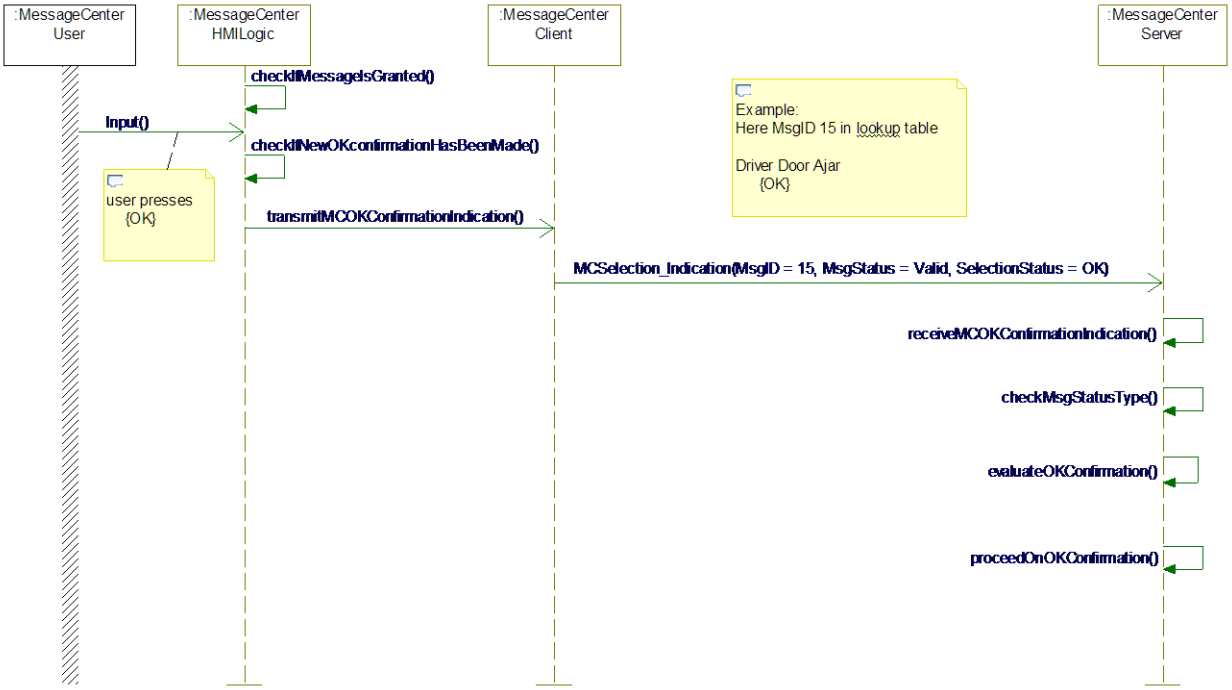
Pre-condition

MC notification is granted.

Post-condition

This message is still presented to the user.

Sequence Diagram



4.3.4.2 MCv3-SD-REQ-347381/A-Message Center Notification Confirmation_showing interaction behavior with another granted message

Scenarios

Normal Usage

A message center notification is triggered (here: MsgID 73 - Rear Park Aid Deactivated). After a short time, another message center notification with an OK-Button is triggered (here: MsgID 15 - Driver Door Ajar).
User presses OK to confirm notification.
MessageCenter Server first brings the notification in the back into front (via "update") and then deactivates the confirmed one to avoid flickering.

Constraints

Pre-condition

System is on.

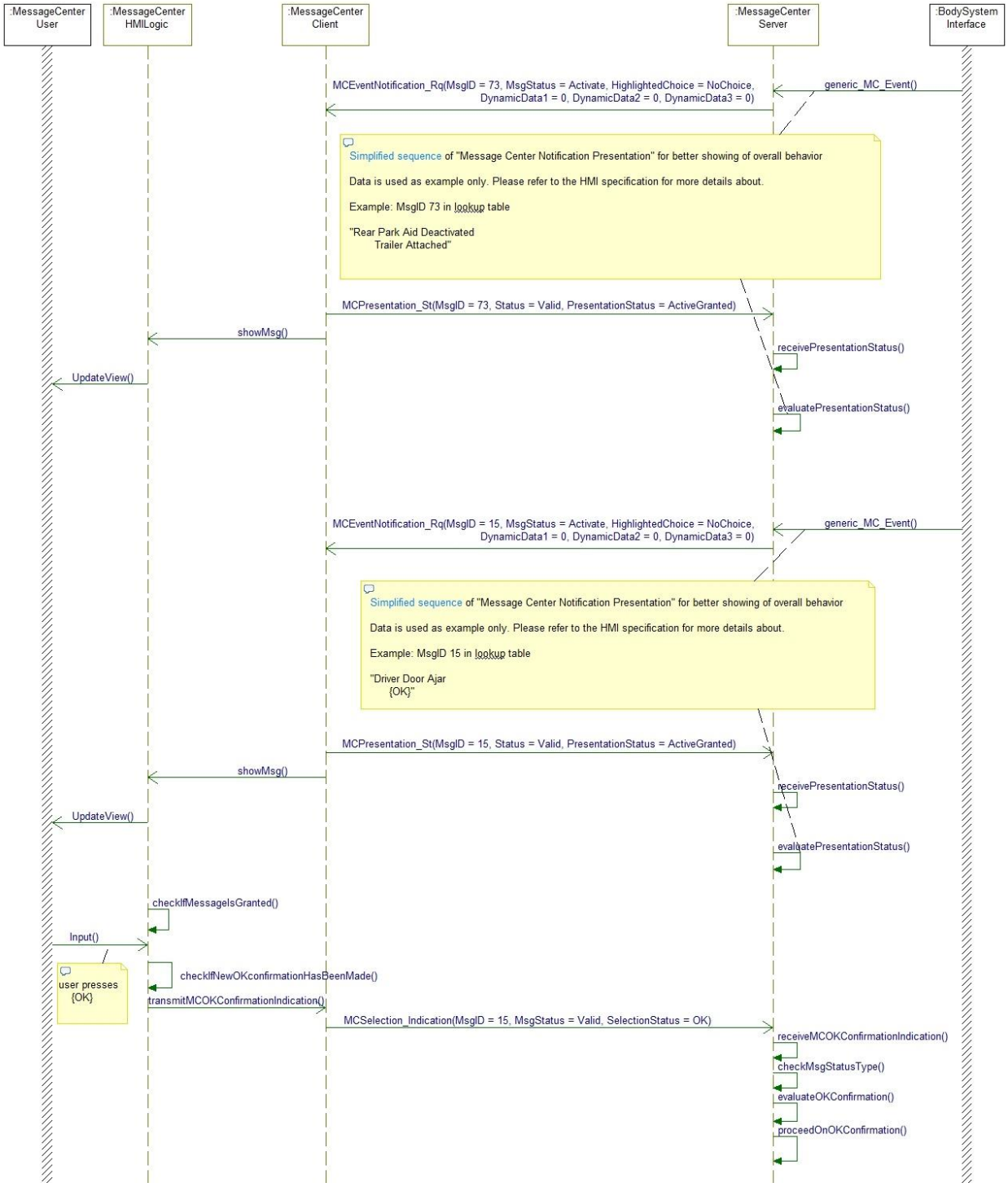
Pre-condition

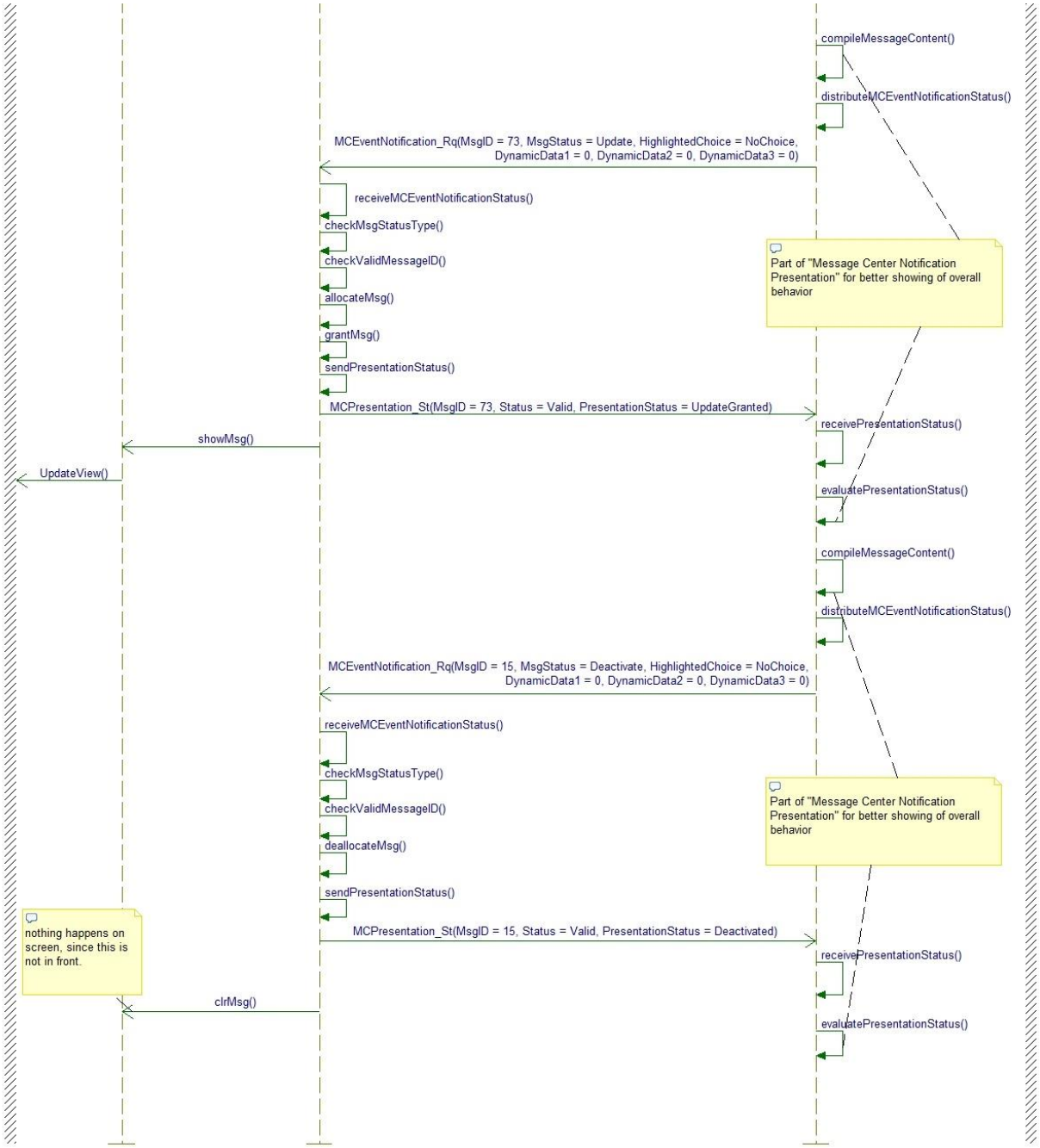
No MC notification is active.

Post-condition

First MC notification is presented to the user.

Sequence Diagram




compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 73, MsgStatus = Update, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
allocateMsg()
grantMsg()
sendPresentationStatus()
Part of "Message Center Notification Presentation" for better showing of overall behavior
MCPresentation_St(MsgID = 73, Status = Valid, PresentationStatus = UpdateGranted)
receivePresentationStatus()
showMsg()
UpdateView()
evaluatePresentationStatus()
compileMessageContent()
distributeMCEventNotificationStatus()
MCEventNotification_Rq(MsgID = 15, MsgStatus = Deactivate, HighlightedChoice = NoChoice, DynamicData1 = 0, DynamicData2 = 0, DynamicData3 = 0)
receiveMCEventNotificationStatus()
checkMsgStatusType()
checkValidMessageID()
deallocateMsg()
Part of "Message Center Notification Presentation" for better showing of overall behavior
sendPresentationStatus()
nothing happens on screen, since this is not in front.
MCPresentation_St(MsgID = 15, Status = Valid, PresentationStatus = Deactivated)
receivePresentationStatus()
clrMsg()
evaluatePresentationStatus()

4.4 MCv3-FUN-REQ-347031/A-Message Center Notification Choice Selection

4.4.1 Requirements

4.4.2 Use Cases

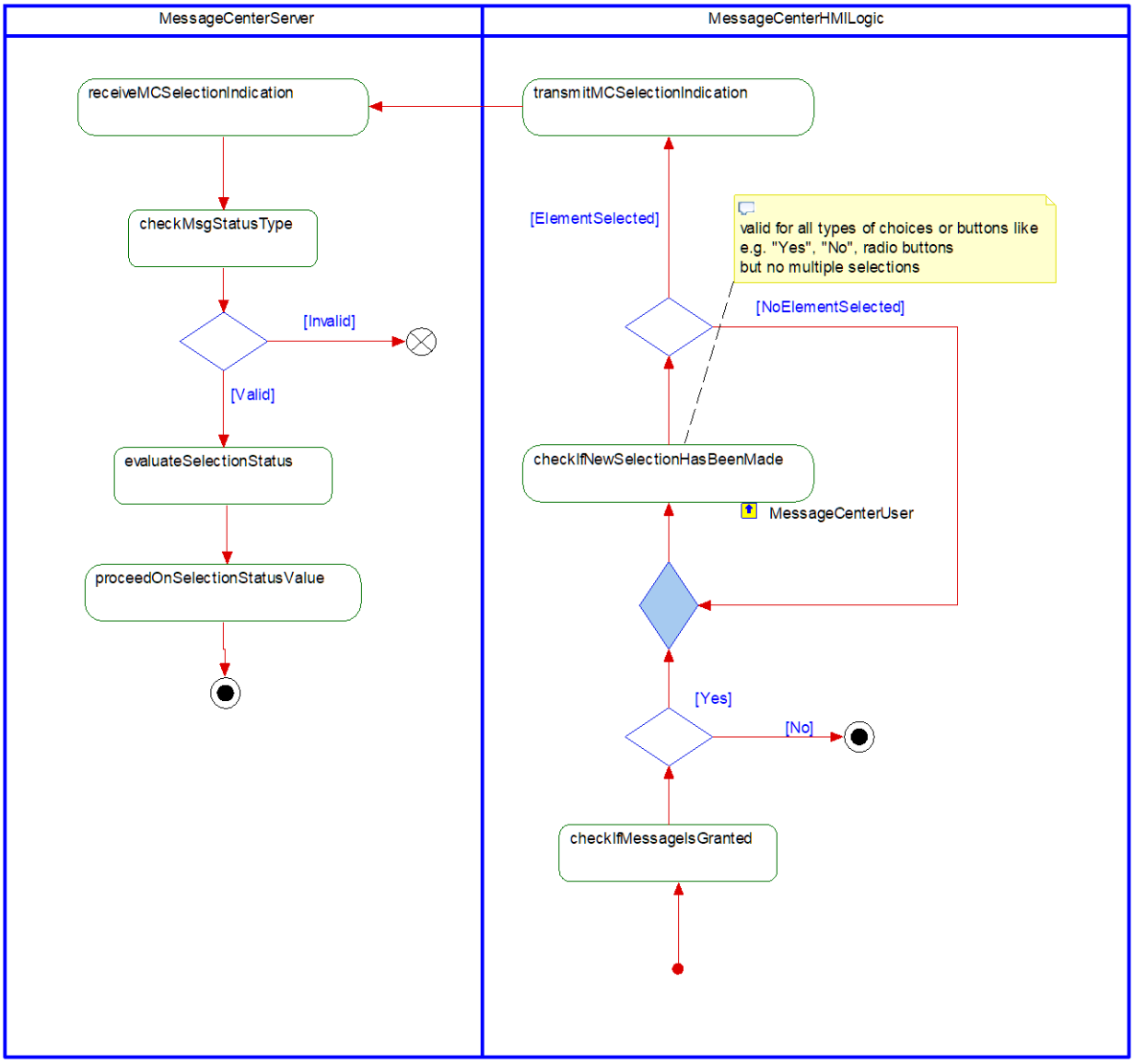
4.4.2.1 MCv3-UC-REQ-347033/A-Message Center Notification Choice Selection

Actors	Message Center User
Pre-conditions	A message center notification with at least one choice element (e.g. “yes” button) is presented to the user (granted).
Scenario Description	User presses one choice element (e.g. “yes” button)
Post-conditions	MessageCenter Client sends an information to MessageCenter Server which number of choice element has been pressed by the user.
List of Exception Use Cases	NA
Interfaces	G-HMI MCConfirmationSelection_Ind

4.4.3 Activity Diagrams

4.4.3.1 MCv3-ACT-REQ-347034/A-Message Center Notification Choice Selection

Activity Diagram



4.4.4 Sequence Diagrams

4.4.4.1 MCv3-SD-REQ-347040/A-Message Center Notification Choice Selection_Messages available, User Choice 1 Selection, Successor Msg not available

Scenarios

Normal Usage

In this example, User selects choice 1 (here: Yes). MessageCenter Client compiles selection indication message with choice 1 selected and sends to MessageCenter Server.

This checks validity (correct message type and ID) evaluates that choice 1 has been selected and means “Yes” in this case and proceeds accordingly.

Constraints

Pre-condition

System is on.

Pre-condition

Notification with at least one choice element is presented to the user.

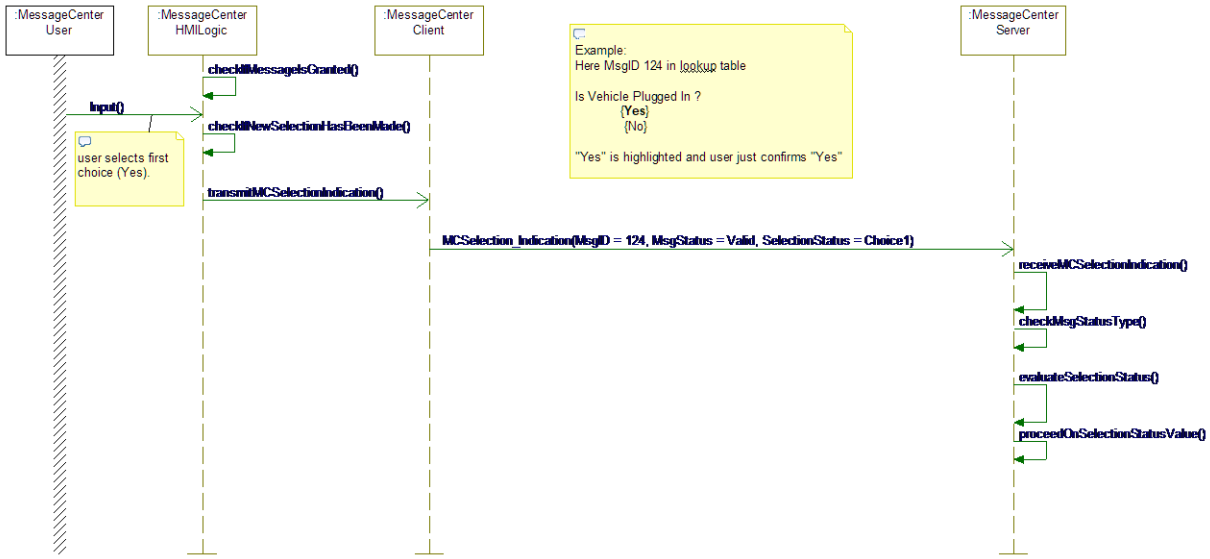
Post-condition

This MC notification is still presented to the user.

Post-condition

MessageCenter Client sends a confirmation indication information is sent to MessageCenter Server to transfer choice 1 has been selected by user.

Sequence Diagram



4.5 MCv3-FUN-REQ-347021/A-Dynamic Data Gathering

4.5.1 Requirements

4.5.2 Use Cases

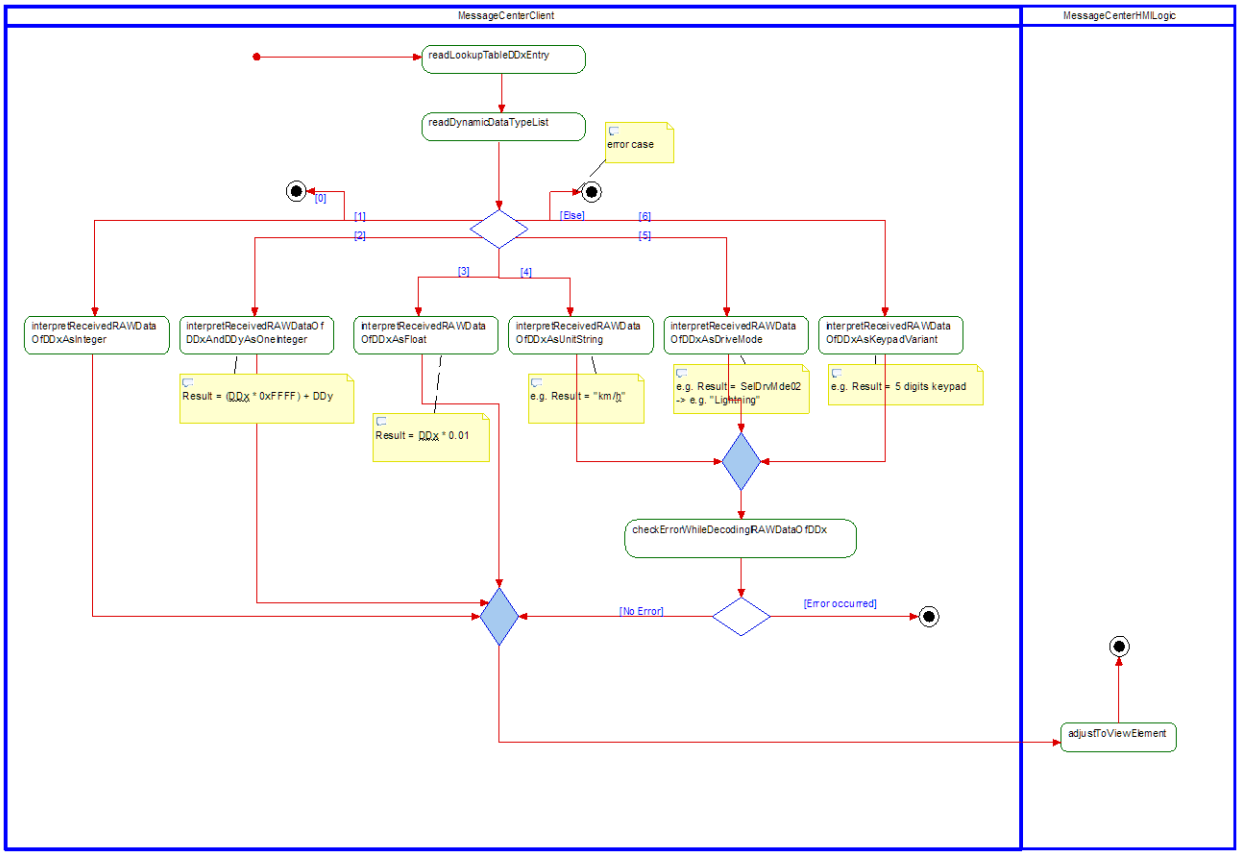
4.5.2.1 MCv3-UC-REQ-347023/A-Dynamic Data Gathering

Actors	Message Center System
Pre-conditions	A notification message has been received and validity checks are successful.
Scenario Description	MessageCenter Client gets type of dynamic data field from lookup table and looks in DD type list how to interpret content of this/these raw data field/s and formats it for related view element.
Post-conditions	Content of dynamic data field is interpreted and in correct format for view element.
List of Exception Use Cases	NA
Interfaces	Message Center System

4.5.3 Activity Diagrams

4.5.3.1 MCv3-ACT-REQ-347024/A-Dynamic Data Gathering

Activity Diagram



4.5.4 Sequence Diagrams

4.5.4.1 MCv3-SD-REQ-347025/A-Dynamic Data Type is Integer

Scenarios

Normal Usage

MessageCenter Client gets type of dynamic data field from lookup table and looks in DD type list to see to interpret content of this raw data field as Integer and formats it for related view element.

Constraints

Pre-condition

System is on.

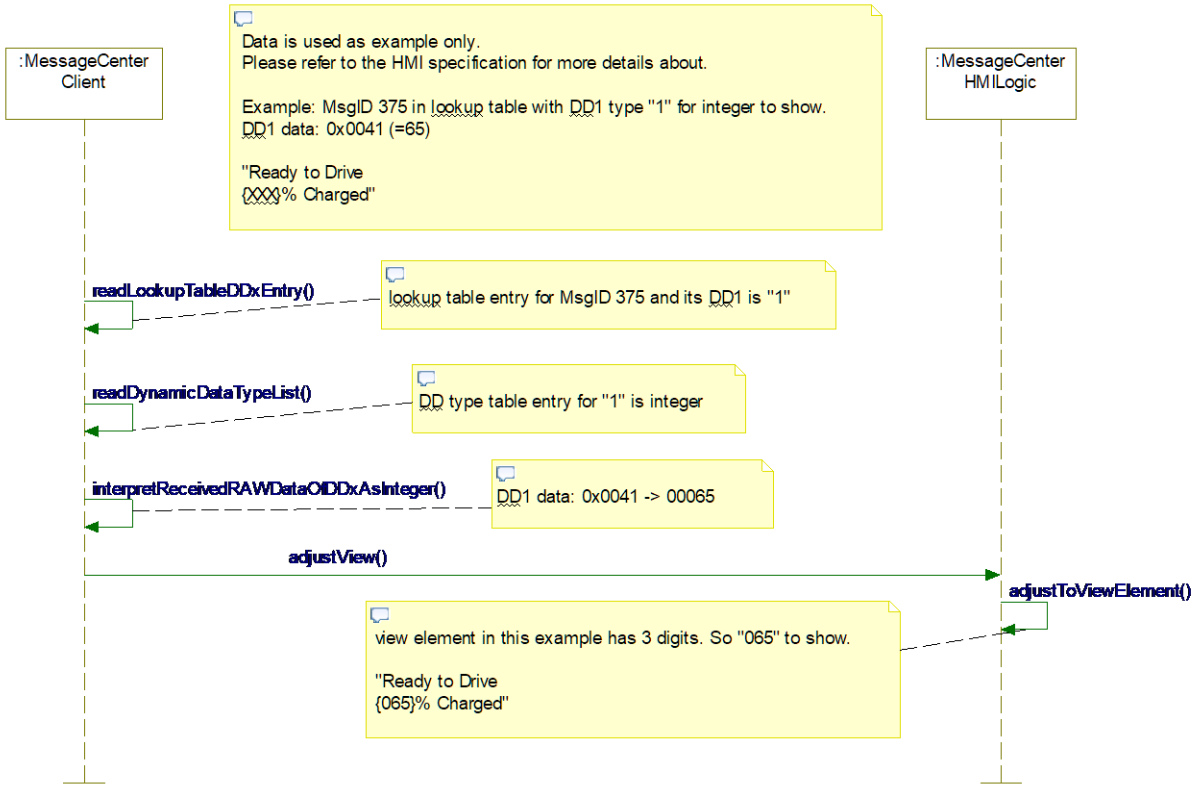
Pre-condition

A notification message has been received and validity checks are successful.

Post-condition

Content of dynamic data field is interpreted as Integer value and in correct format for view element.

Sequence Diagram



4.5.4.2 MCv3-SD-REQ-348792/A-Dynamic Data Type is two as one Integer

Scenarios

Normal Usage

MessageCenter Client gets type of dynamic data field from lookup table and looks in DD type list to see to interpret content of two raw data fields as one Integer and formats it for related view element.

Constraints

Pre-condition

System is on.

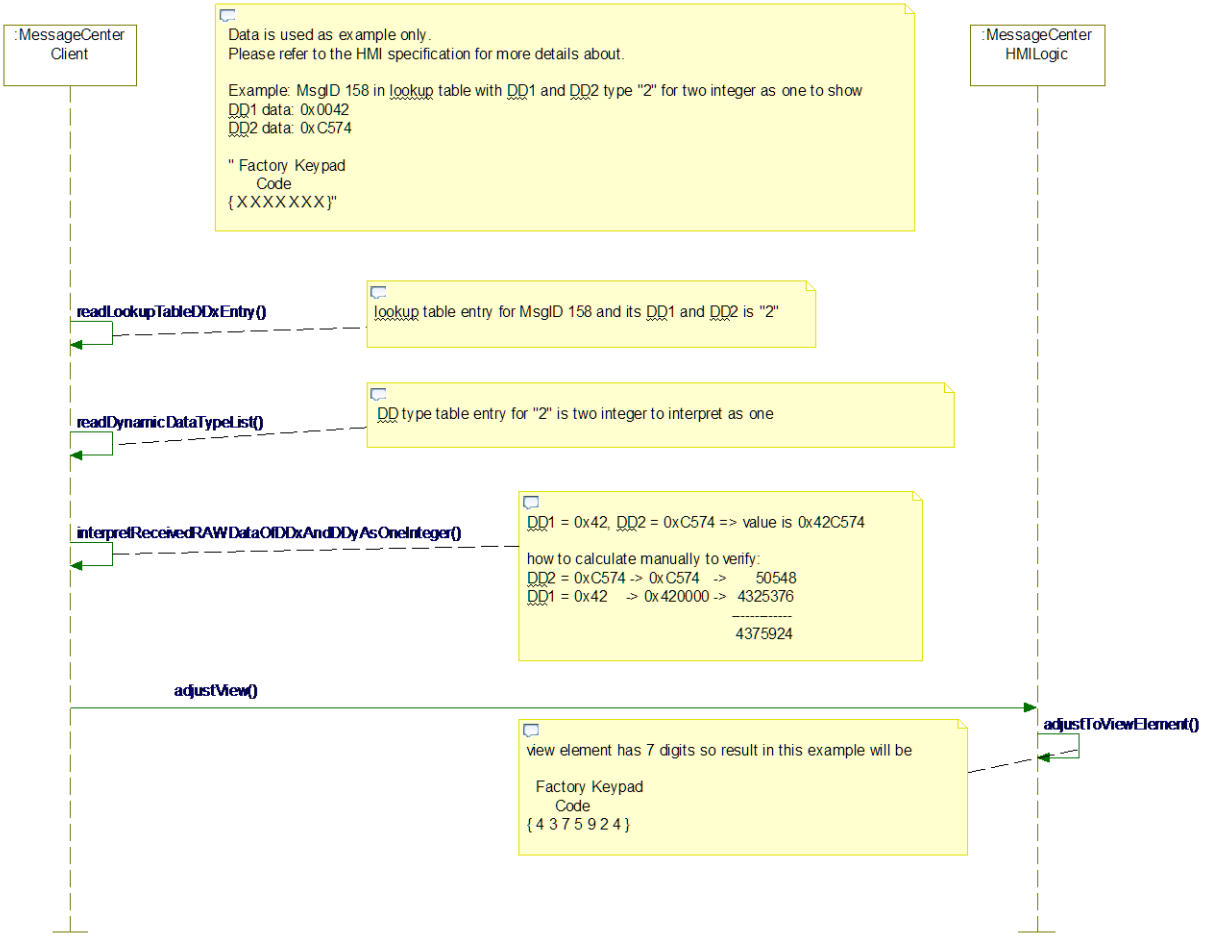
Pre-condition

A notification message has been received and validity checks are successful.

Post-condition

Content of two dynamic data fields are interpreted as one Integer value and in correct format for view element.

Sequence Diagram



4.5.4.3 MCv3-SD-REQ-348793/A-Dynamic Data Type is Float

Scenarios

Normal Usage

MessageCenter Client gets type of dynamic data field from lookup table and looks in DD type list to see to interpret content of this raw data field as Float, multiply with 0.01 and formats it for related view element.

Constraints

Pre-condition

System is on.

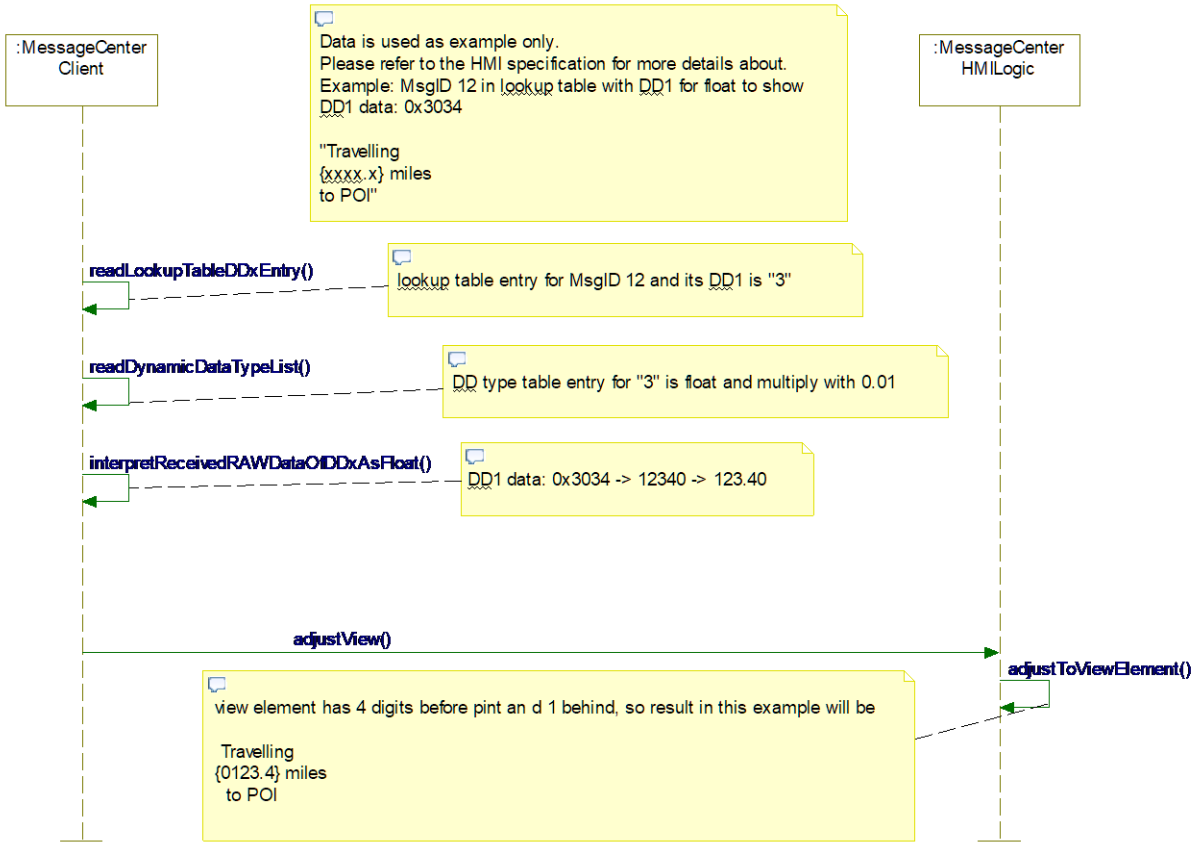
Pre-condition

A notification message has been received and validity checks are successful.

Post-condition

Content of dynamic data field is interpreted as Float value and in correct format for view element.

Sequence Diagram



4.5.4.4 MCv3-SD-REQ-348794/A-Dynamic Data Type is Unit

Scenarios

Normal Usage

MessageCenter Client gets type of dynamic data field from lookup table and looks in DD type list to see to interpret content of this raw data field as unit enumeration literal for a unit string and formats it for related view element.

Constraints

Pre-condition

System is on.

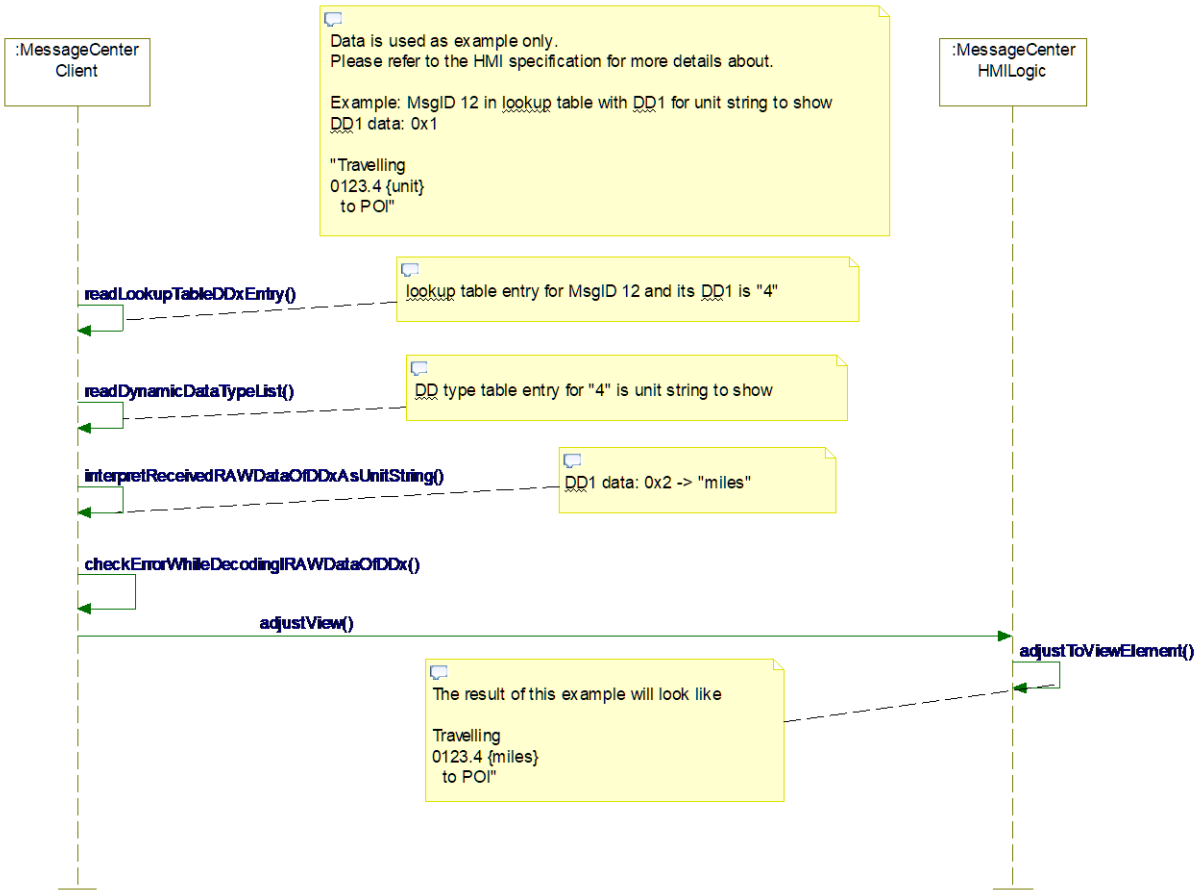
Pre-condition

A notification message has been received and validity checks are successful.

Post-condition

Content of dynamic data field is interpreted as unit string value and in correct format for view element.

Sequence Diagram



4.5.4.5 MCv3-SD-REQ-348795/A-Dynamic Data Type is Drive Mode

Scenarios

Normal Usage

MessageCenter Client gets type of dynamic data field from lookup table and looks in DD type list to see to interpret content of this raw data field as drive mode enumeration literal for a drive mode string and formats it for related view element.

Constraints

Pre-condition

System is on.

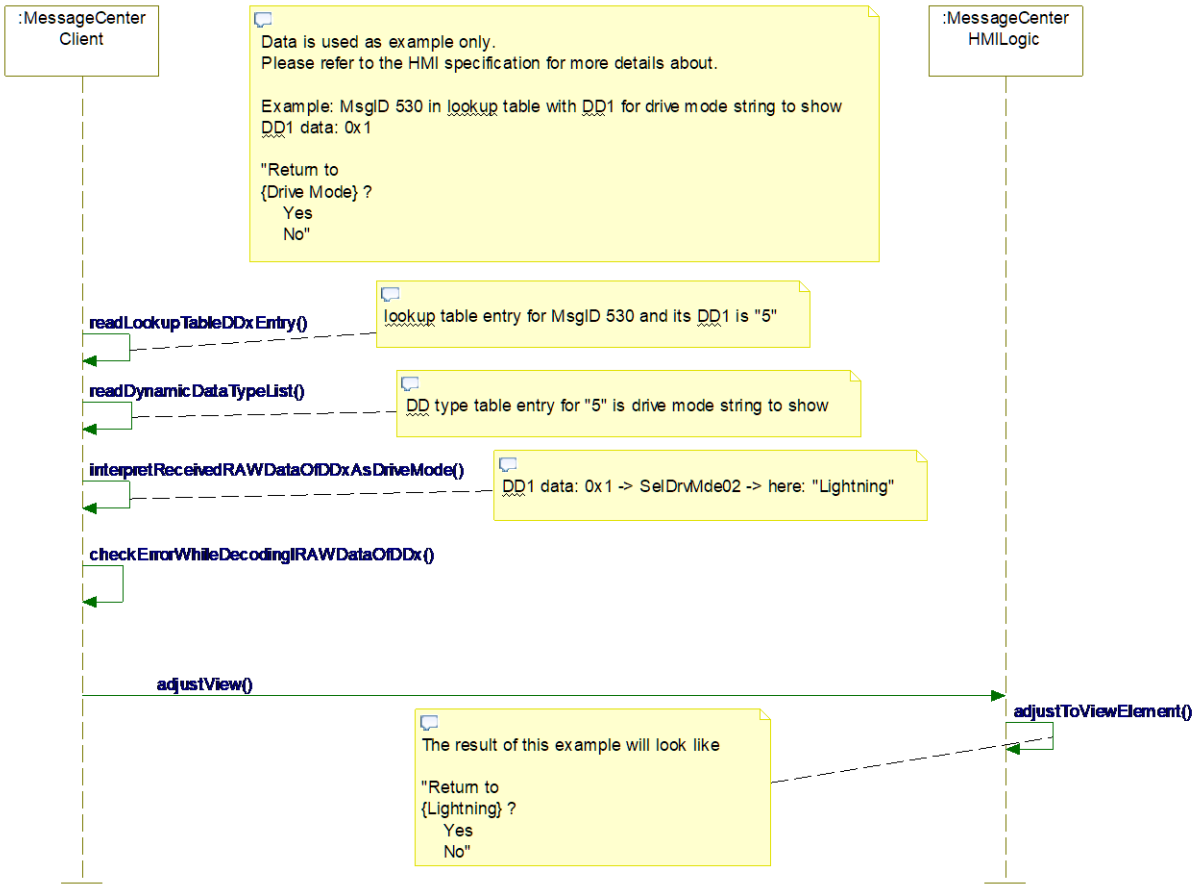
Pre-condition

A notification message has been received and validity checks are successful.

Post-condition

Content of dynamic data field is interpreted as drive mode string value and in correct format for view element.

Sequence Diagram



4.5.4.6 MCv3-SD-REQ-348796/A-Dynamic Data Type is Keypad digits

Scenarios

Normal Usage

MessageCenter Client gets type of dynamic data field from lookup table and looks in DD type list to see to interpret content of this raw data field as digit count enumeration literal for showing correct number of digits for keypad.

Constraints

Pre-condition

System is on.

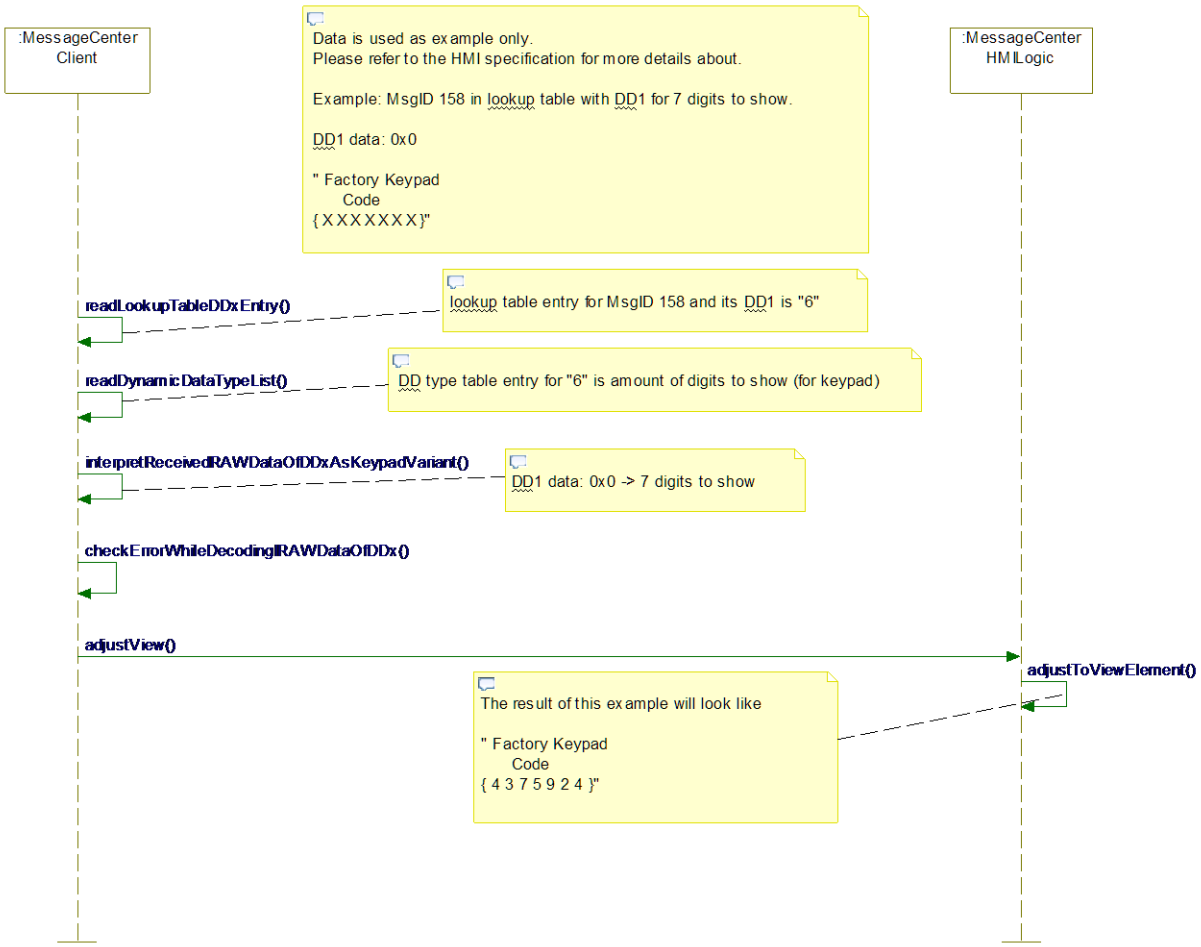
Pre-condition

A notification message has been received and validity checks are successful.

Post-condition

Content of dynamic data field is interpreted as digit count enumeration literal for showing correct number of digits for keypad and in correct format for view element.

Sequence Diagram



4.5.4.7 MCv3-SD-REQ-348797/A-Dynamic Data Type is Not Used

Scenarios

Normal Usage

MessageCenter Client gets type of dynamic data field from lookup table and looks in DD type list to see to NOT interpret content of this raw data, since it is of type “not used”.

Constraints

Pre-condition

System is on.

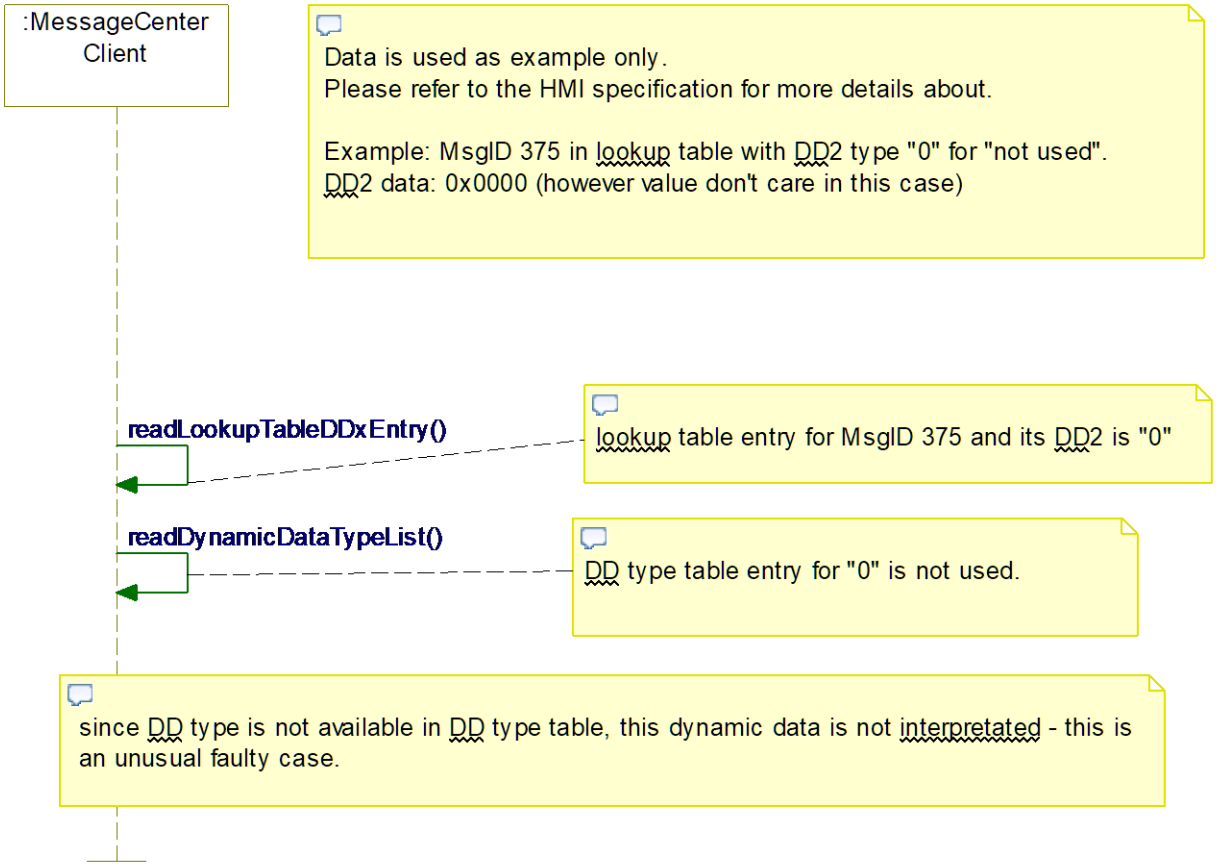
Pre-condition

A notification message has been received and validity checks are successful.

Post-condition

Content of dynamic data field is interpreted as “not used” and ignored.

Sequence Diagram



4.5.4.8 MCv3-SD-REQ-348798/A-Dynamic Data Type is Not Available

Scenarios

Normal Usage

MessageCenter Client gets type of dynamic data field from lookup table and looks in DD type list to see this DD type is not available.
So it does not interpret content and react like in “MCv3-SD-REQ-349024-Message Center Notification Presentation_Error occurred while RAW Data Decoding”.

Constraints

Pre-condition

System is on.

Pre-condition

A notification message has been received and validity checks are successful.

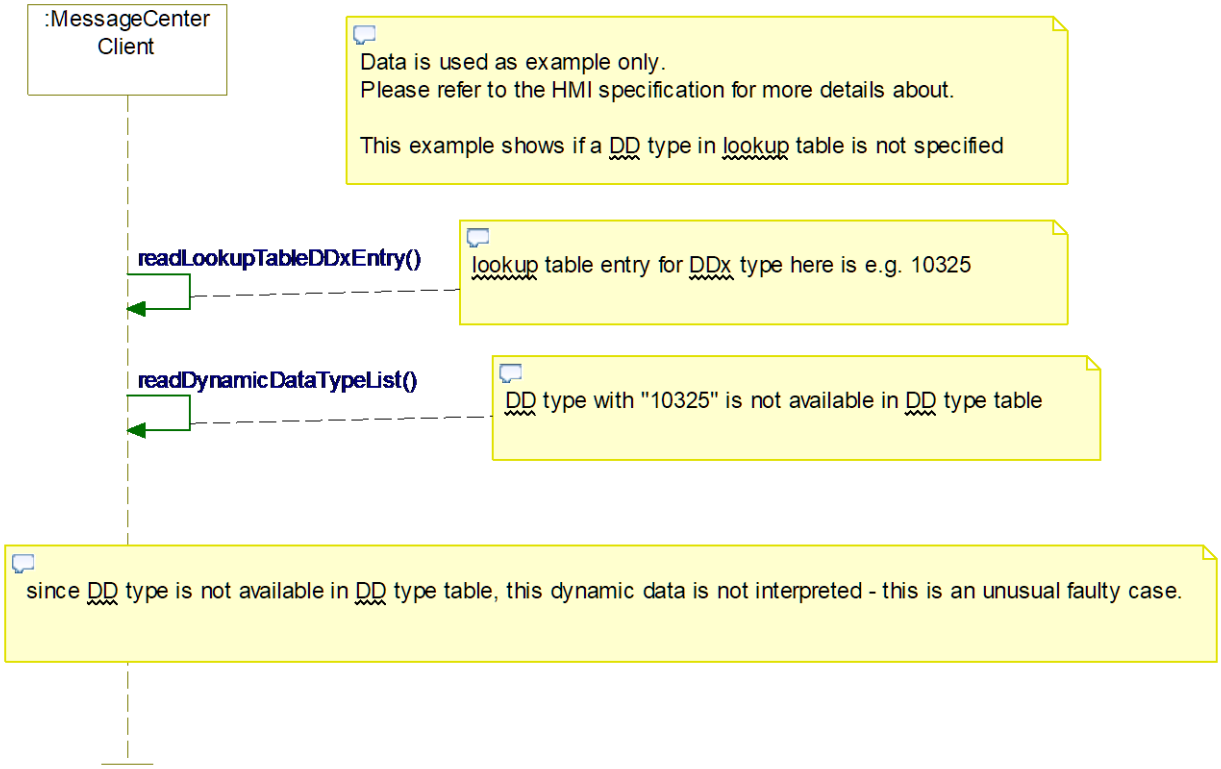
Post-condition

Content of dynamic data field is ignored since DD type could not be found in DD type list.

Post-condition

MessageCenter Client reacts like described in “MCv3-SD-REQ-349024-Message Center Notification Presentation_Error occurred while RAW Data Decoding”.

Sequence Diagram



4.5.4.9 MCv3-SD-REQ-349033/A-Dynamic Data Type is Not possible to decode

Scenarios

Normal Usage

MessageCenter Client gets type of dynamic data field from lookup table and looks in DD type list to see to interpret content of this raw data field as unit enumeration literal for a unit string.
However this value of content is not available.
So it does not interpret content and react like in “MCv3-SD-REQ-349024-Message Center Notification Presentation_Error occurred while RAW Data Decoding”.

Constraints

Pre-condition

System is on.

Pre-condition

A notification message has been received and validity checks are successful.

Post-condition

Content of dynamic data field is ignored since value content of this DD type could not be found in DD type list.

Post-condition

MessageCenter Client reacts like described in “MCv3-SD-REQ-349024-Message Center Notification Presentation_Error occurred while RAW Data Decoding”.

Sequence Diagram

4.6 MCv3-FUN-REQ-347869/A-Message Center Deallocate All Messages At Once

4.6.1 Requirements

4.6.2 Use Cases

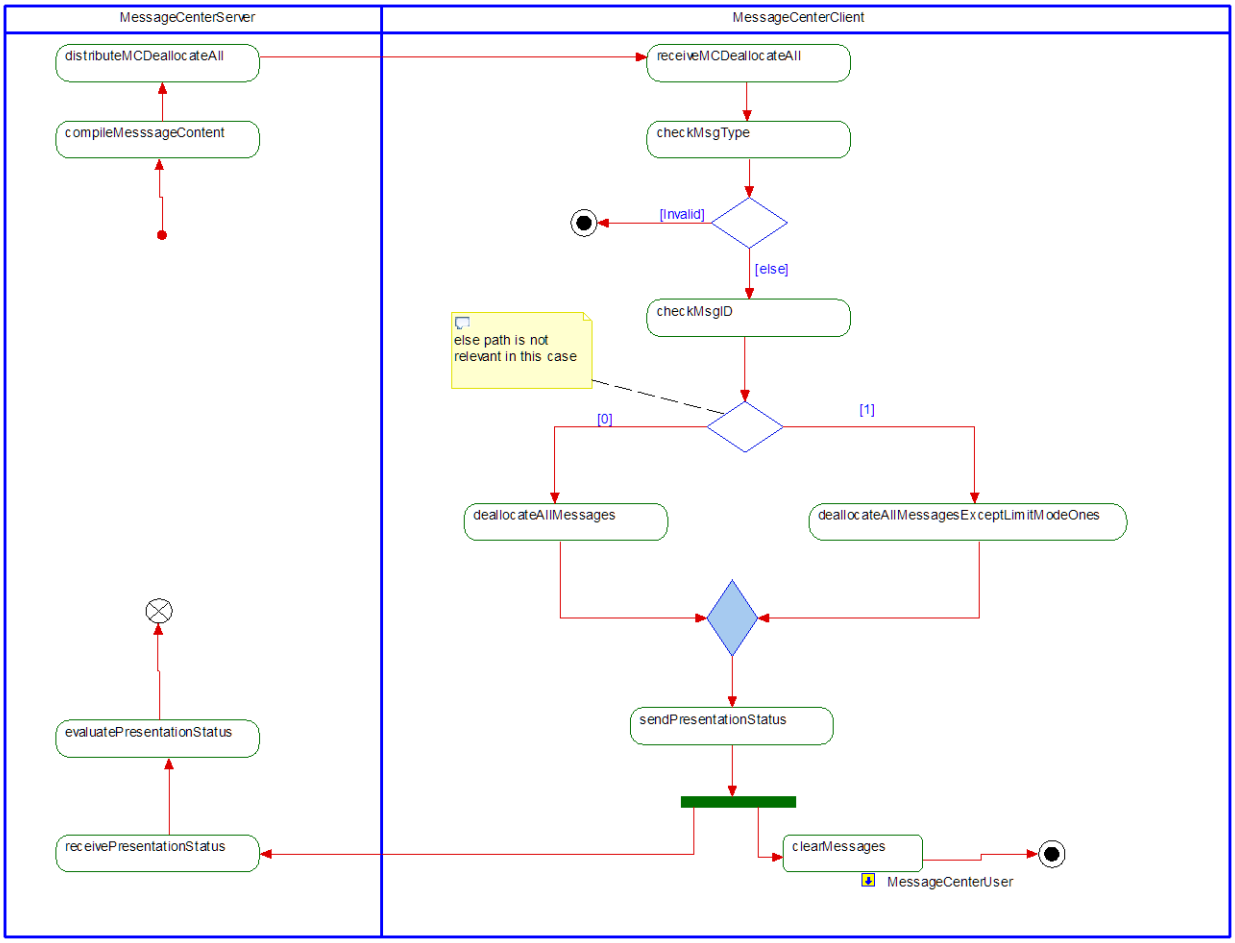
4.6.2.1 UC-REQ-347871/A-Message Center Deallocate All Messages At Once

Actors	Message Center System
Pre-conditions	MessageCenter Server needs to deallocate/deactivate all messages at once
Scenario Description	MessageCenter Server compiles related message content and transmits this notification message to MessageCenter Client. This checks validity (correct message type and ID) and deallocates/deactivates all message or all messages except limit mode ones and sends according status back to MessageCenter Server
Post-conditions	All message center notifications (maybe except limit mode ones) are deallocated/deactivated.
List of Exception Use Cases	NA
Interfaces	G-HMI MCConfirmationSelection_Ind

4.6.3 Activity Diagrams

4.6.3.1 MCv3-ACT-REQ-347872/A-Message Center Deallocate All Messages At Once

Activity Diagram



4.6.4 Sequence Diagrams

4.6.4.1 MCv3-SD-REQ-347873/A-Deallocate All Messages At Once_Deactivate All

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of MsgID 0 and transmits this notification message to MessageCenter Client. It can be activate, update or deactivate.
MessageCenter Client checks validity (correct message type and ID) and deallocates all notifications at once.
Then it sends an inactive presentation status back to MessageCenter Server.

Note: all other data like HighlightedChoice or Dynamic Data fields do not care and shall be ignored.

Constraints

Pre-condition

System is on.

Pre-condition

It does not care whether any message is active.

Pre-condition

Message ID „0“ is sent.

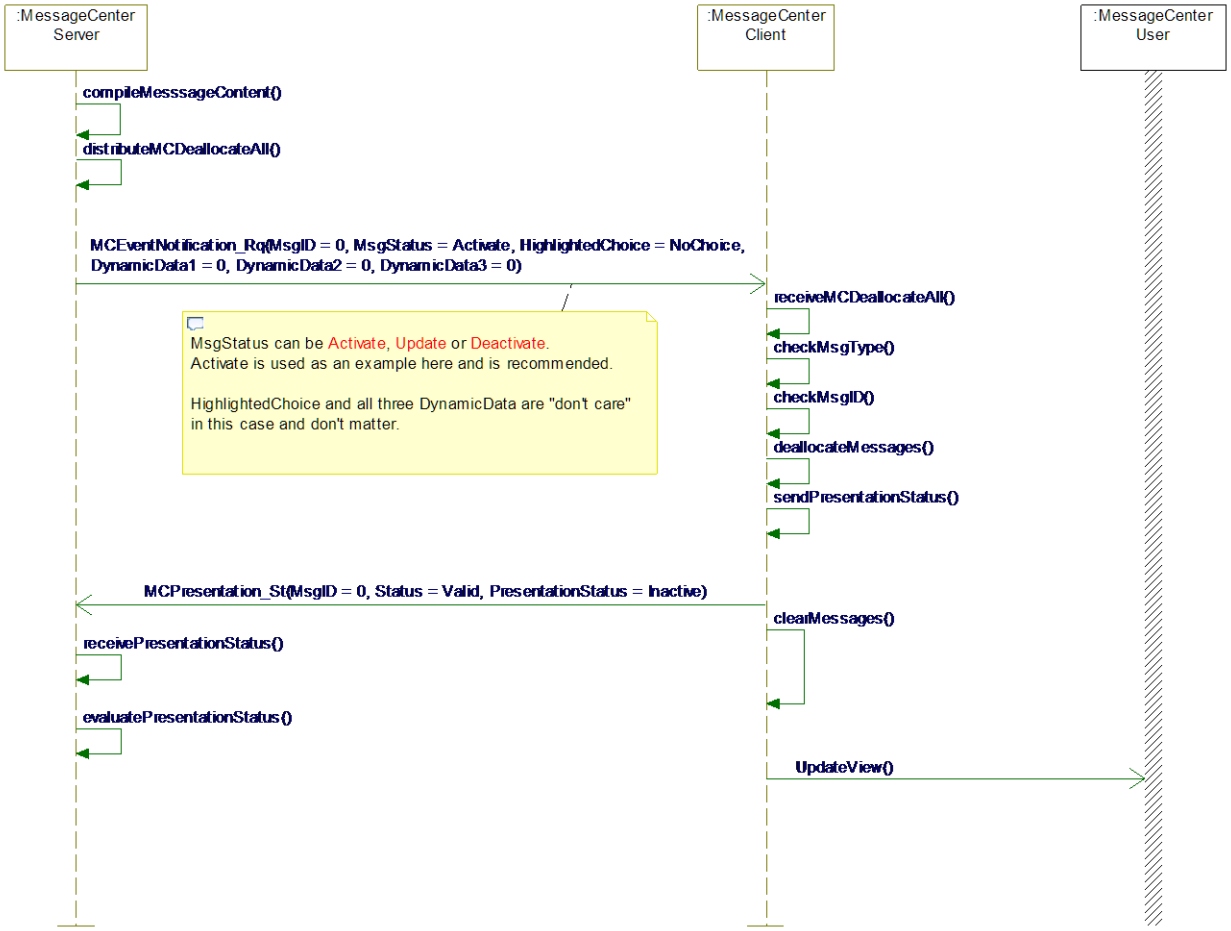
Post-condition

All messages are deallocated. Means stack is empty.

Post-condition

No message is presented to user

Sequence Diagram



4.6.4.2 MCv3-SD-REQ-347874/A-Deallocate All Messages At Once_Deallocate All except Limit Mode Ones

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of MsgID 1 and transmits this notification message to MessageCenter Client. It can be activate, update or deactivate.
MessageCenter Client checks validity (correct message type and ID) and deallocates all notifications at once except limit mode ones. This means, notifications defined to be available in limit mode, will not be deallocated in this case, however all other.

Then it sends an inactive presentation status back to MessageCenter Server.

Note: all other data like HighlightedChoice or Dynamic Data fields do not care and shall be ignored.

Constraints

Pre-condition

System is on.

Pre-condition

It does not care whether any message is active.

Pre-condition

Message ID „1“ is sent.

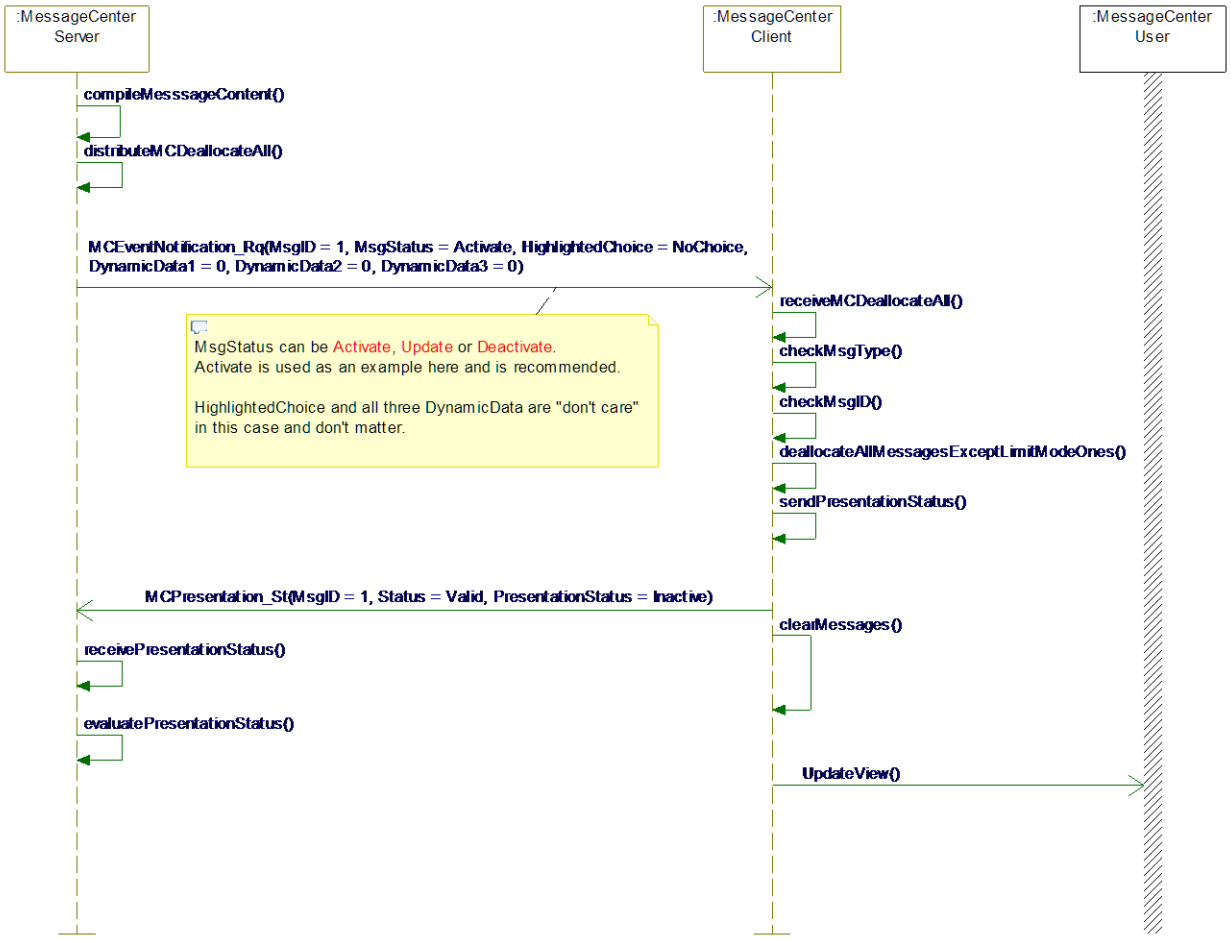
Post-condition

All messages are deallocated. Means stack is empty.

Post-condition

No message is presented to user

Sequence Diagram



4.6.4.3 MCv3-SD-REQ-347875/A-Deallocate All Messages At Once_MessageType Invalid

Scenarios

Normal Usage

In this example, MessageCenter Server compiles message content of any MsgID and transmits this notification message to MessageCenter Client. However, message type is Invalid.
MessageCenter Client checks validity (correct message type and ID) and ignores whole message.

Constraints

Pre-condition

System is on.

Pre-condition

It does not care whether any message is active.

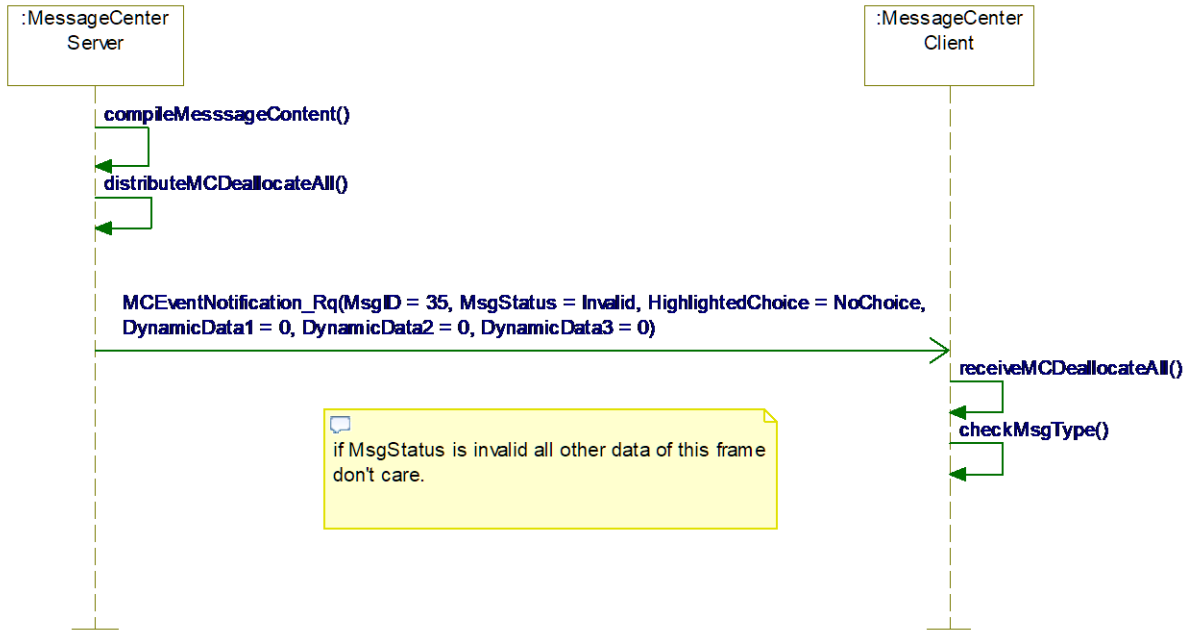
Pre-condition

Message type “Invalid” is sent for intention to empty messages stack.

Post-condition

No change.

Sequence Diagram



4.7 MCv3-FUN-REQ-348224/A-Message Center Global Alert Notification

4.7.1 Requirements

4.7.1.1 MCv3-REQ-347032/A-Message Center Global Alert Notification_StatusType is Activate, no Message in Stack

If MessageCenter Server receives a Global Alert trigger but there is no message granted (no message center notification in stack) it shall not send out a message center notification activation request for global alert to MessageCenter Client.

4.7.2 Use Cases

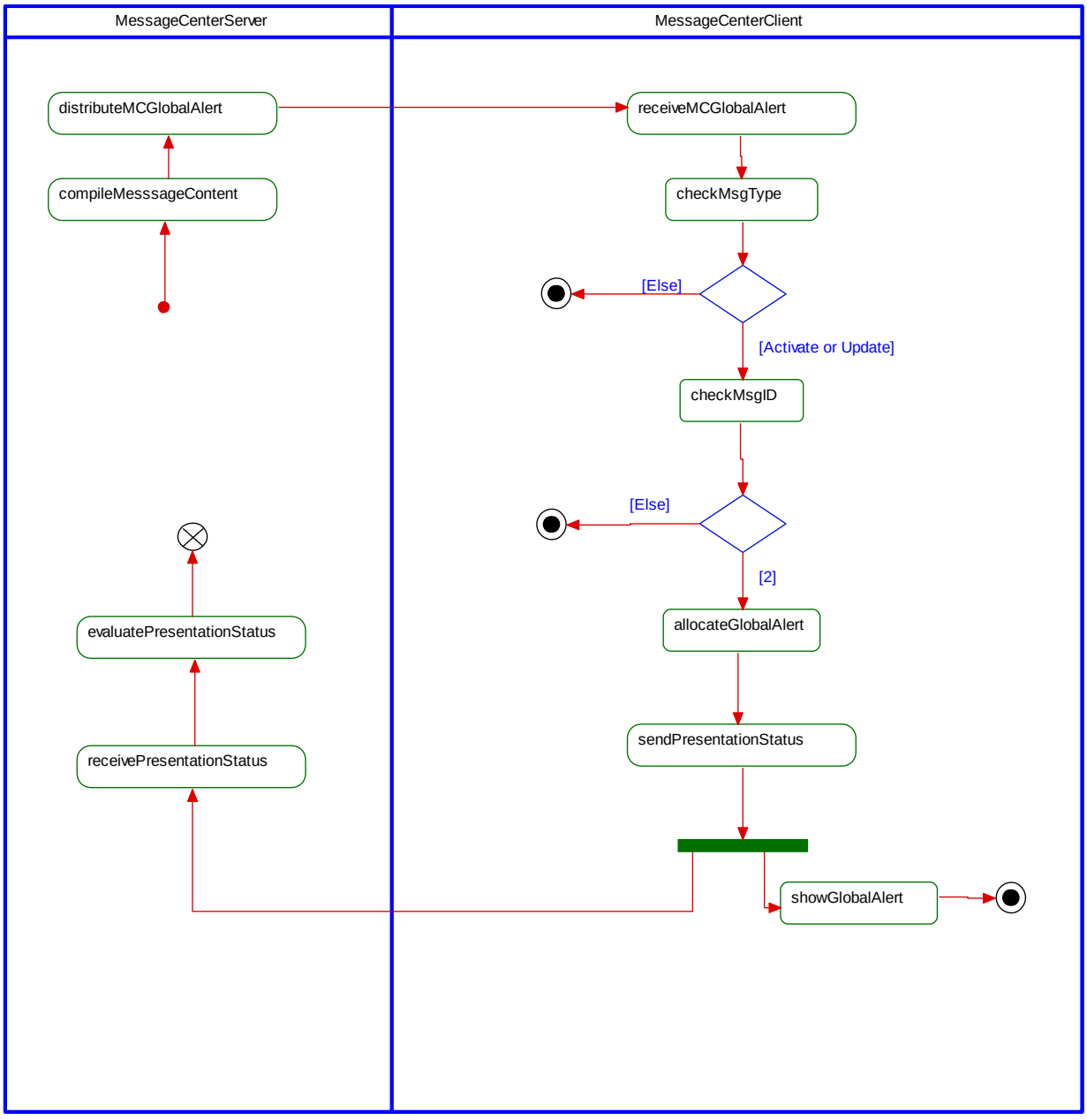
4.7.2.1 MCv3-UC-REQ-348226/A-Message Center Global Alert Notification

Actors	Vehicle System
Pre-conditions	A global alert event occurs or finishes.
Scenario Description	MessageCenter Server compiles related message content and transmits this notification message to MessageCenter Client. This checks validity (correct message type and ID) and activates/deactivates “global alert screen” (-> see HMI specification for further information) and sends according status back to MessageCenter Server.
Post-conditions	Global alert screen activated or deactivated.
List of Exception Use Cases	NA
Interfaces	G-HMI Vehicle System Interface

4.7.3 Activity Diagrams

4.7.3.1 MCv3-ACT-REQ-348227/A-Message Center Global Alert Notification

Activity Diagram



4.7.4 Sequence Diagrams

4.7.4.1 MCv3-SD-REQ-348228/A-Message Center Global Alert Notification_StatusType is Activate

Scenarios

Normal Usage

MessageCenter Server sends a global alert notification to MessageCenter Client.
After MessageCenter Client is checking valid data it allocates this blank screen and sends back related presentation status.

Constraints

Pre-condition

System is on.

Pre-condition

It does not care if any message is active.

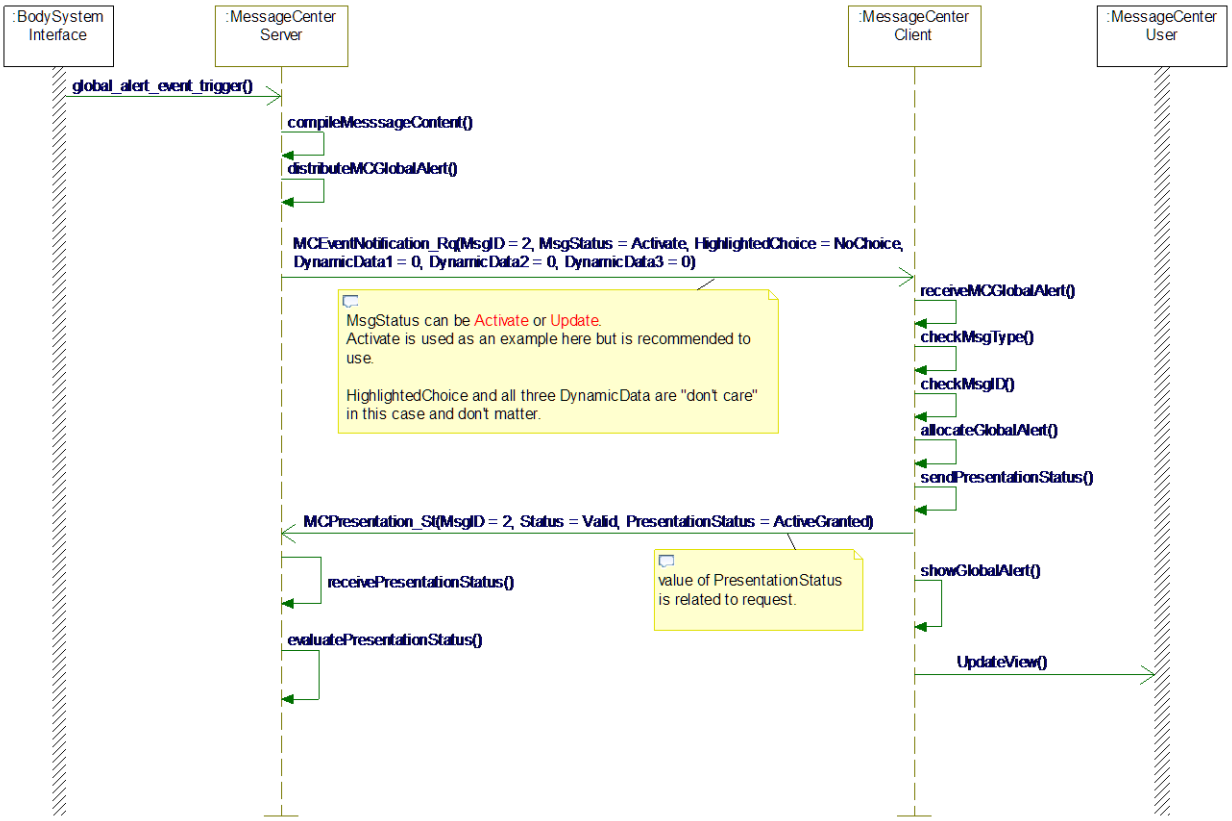
Pre-condition

Message ID „2“ is sent.

Post-condition

A message center notification blank screen is presented to the user.

Sequence Diagram



4.7.4.2 MCv3-SD-REQ-348233/A-Message Center Global Alert Notification_Global Alert event finished,message in stack,common sequence

Scenarios

Normal Usage

MessageCenter Server receives finish of global alert and sends update to old message notification in stack to MessageCenter Client.

After MessageCenter Client is checking validity of the data it reallocates this old message and presents it to the user and sends back related presentation status and removes global alert notification.

Constraints

Pre-condition

System is on.

Pre-condition

It does not care if any message is active.

Pre-condition

Message ID „2“ is sent.

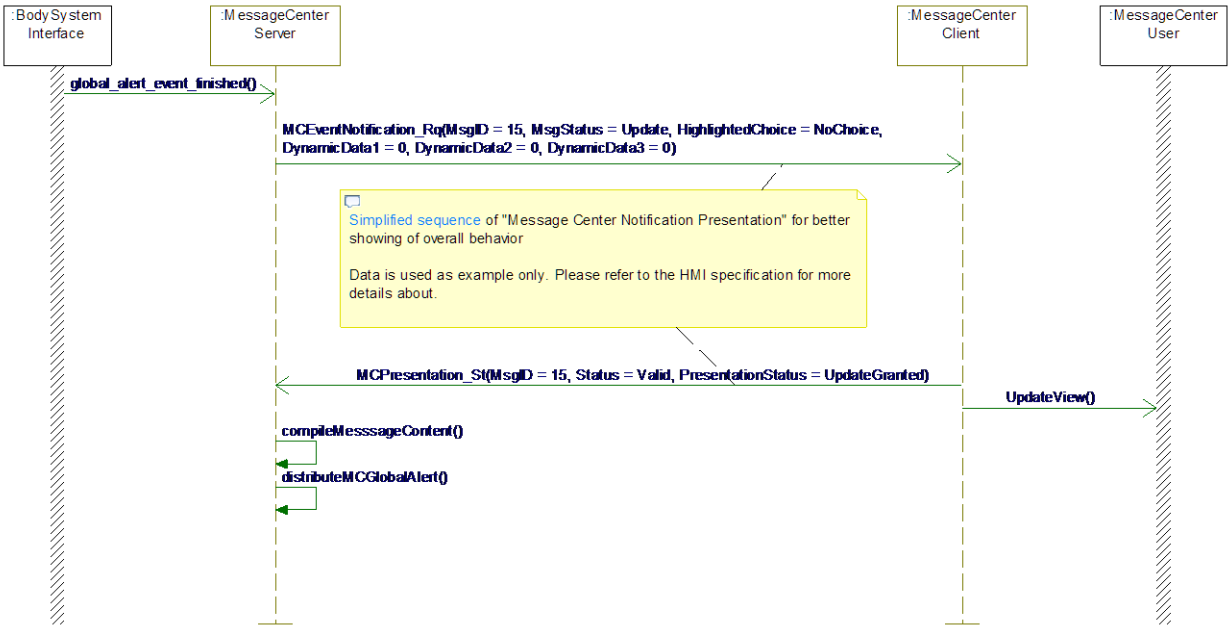
Post-condition

The updated message center notification is presented to the user.

Post-condition

The global alert notification has been removed.

Sequence Diagram



4.7.4.3 MCv3-SD-REQ-348252/A-Message Center Global Alert Notification_StatusType is Invalid or Deactivate

Scenarios

Normal Usage

MessageCenter Server sends a notification request with invalid message type to MessageCenter Client.

MessageCenter Client is checking if message type and message ID are valid. In this case, message type is invalid and MessageCenter Client only ignores whole request.

Constraints

Pre-condition

System is on.

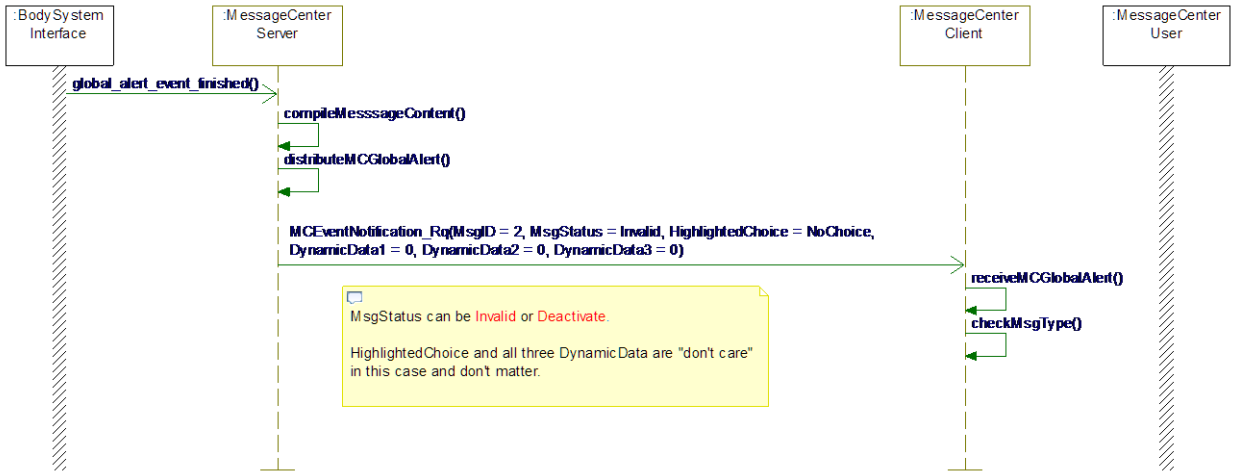
Pre-condition

MessageCenter Server sends message type “Invalid” to MessageCenter Client.

Post-condition

Message notification request is ignored by MessageCenter Client.

Sequence Diagram



4.8 MCv3-FUN-REQ-362264/A-Message Center Broadcast Notification Capability Status

4.8.1 Requirements

4.8.2 Use Cases

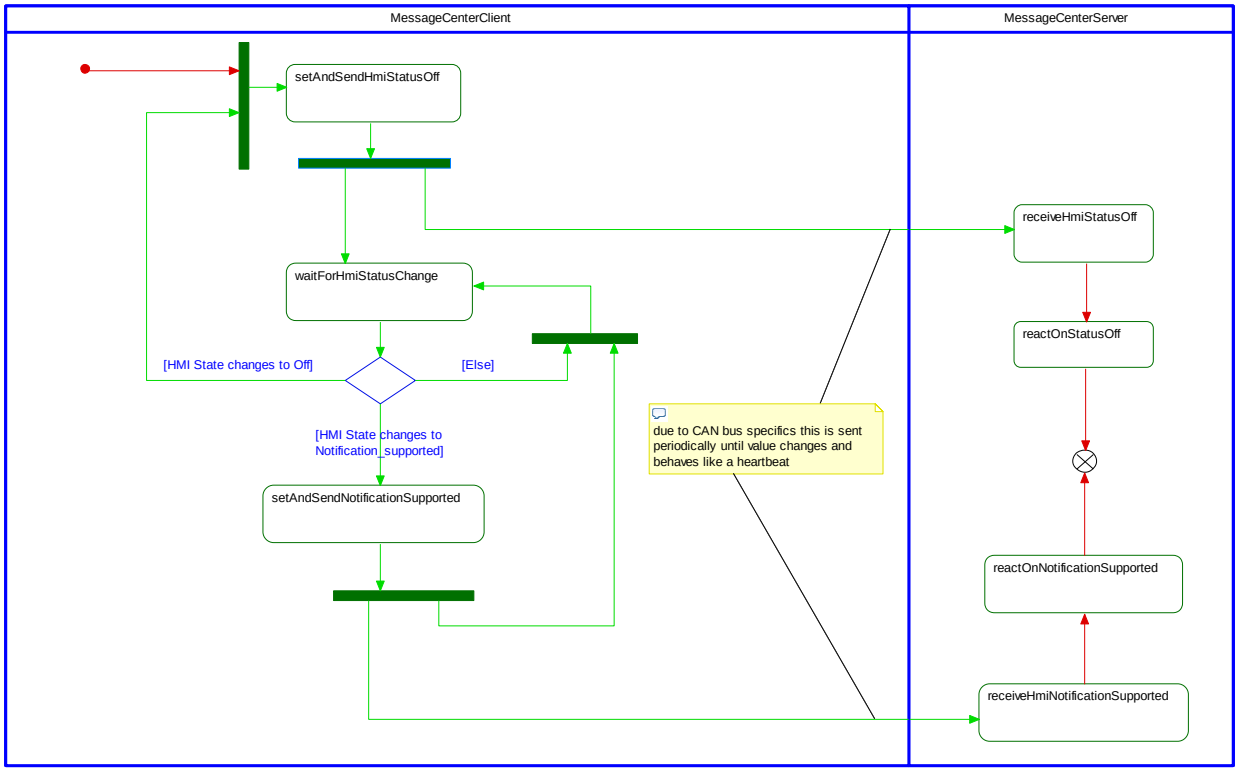
4.8.2.1 MCv3-UC-REQ-362265/A-Broadcast Center Stack HMI Notification Capability

Actors	Center Stack HMI
Pre-conditions	System is capable to transmit messages on bus system
Scenario Description	CenterStackHmiNotification status signal reflects actual HMI status regarding presenting notification status to the user all the time in manner of a heartbeat.
Post-conditions	CenterStackHmiNotification status signal reflects actual HMI notification status and is transmitted periodically
List of Exception Use Cases	None
Interfaces	CenterStackHmiNotification_St

4.8.3 Activity Diagrams

4.8.3.1 MCv3-ACT-REQ-362266/A-Broadcast Center Stack HMI Notification Capability

Activity Diagram



4.8.4 Sequence Diagrams

4.8.4.1 MCv3-SD-REQ-362267/A-Broadcast Center Stack HMI Notification Capability at Startup

Scenarios

Normal Usage

CenterStackHmiNotification status signal is updated with “Notification not capable” and transmitted periodically.

Constraints

Pre-condition

System is booting up or restarting.

Pre-condition

System is capable to transmit messages on bus system.

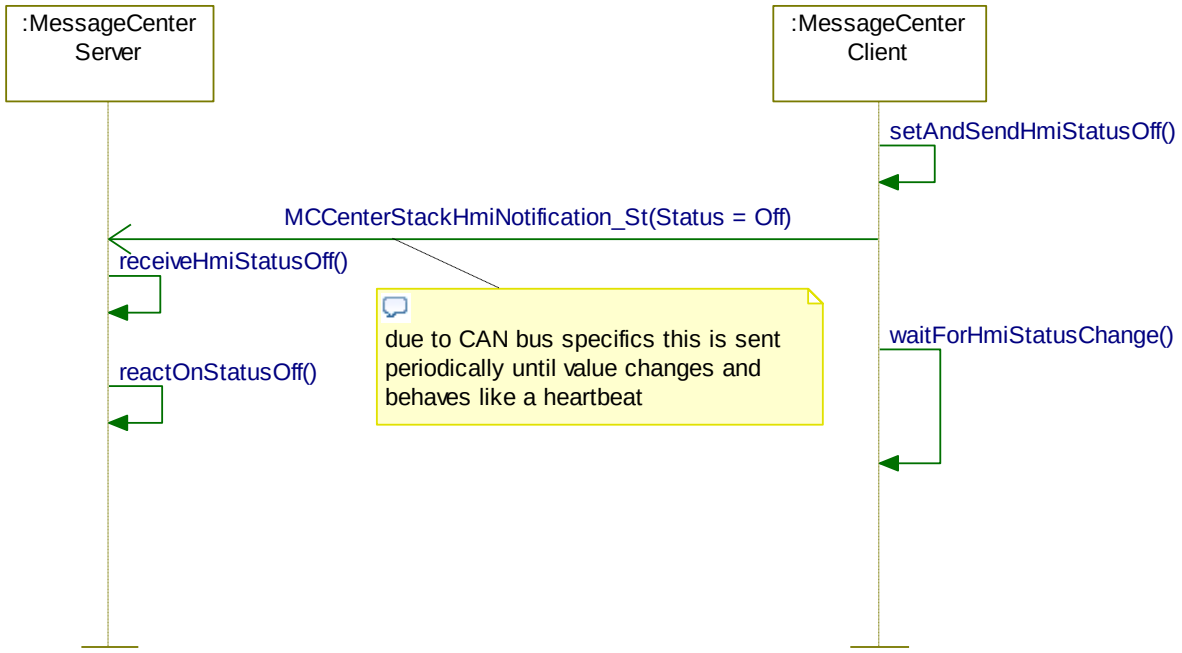
Post-condition

Client is waiting for change of HMI notification status.

Post-condition

CenterStackHmiNotification status signal is transmitted periodically using this value.

Sequence Diagram



4.8.4.2 MCv3-SD-REQ-362268/A-Broadcast Center Stack HMI Notification Capability changes to On

Scenarios

Normal Usage

HMI notification status changes, since HMI is now capable to present notifications to the user. CenterStackHmiNotification status signal is updated with “Notification capable” and transmitted periodically.

Constraints

Pre-condition

System is up and running.

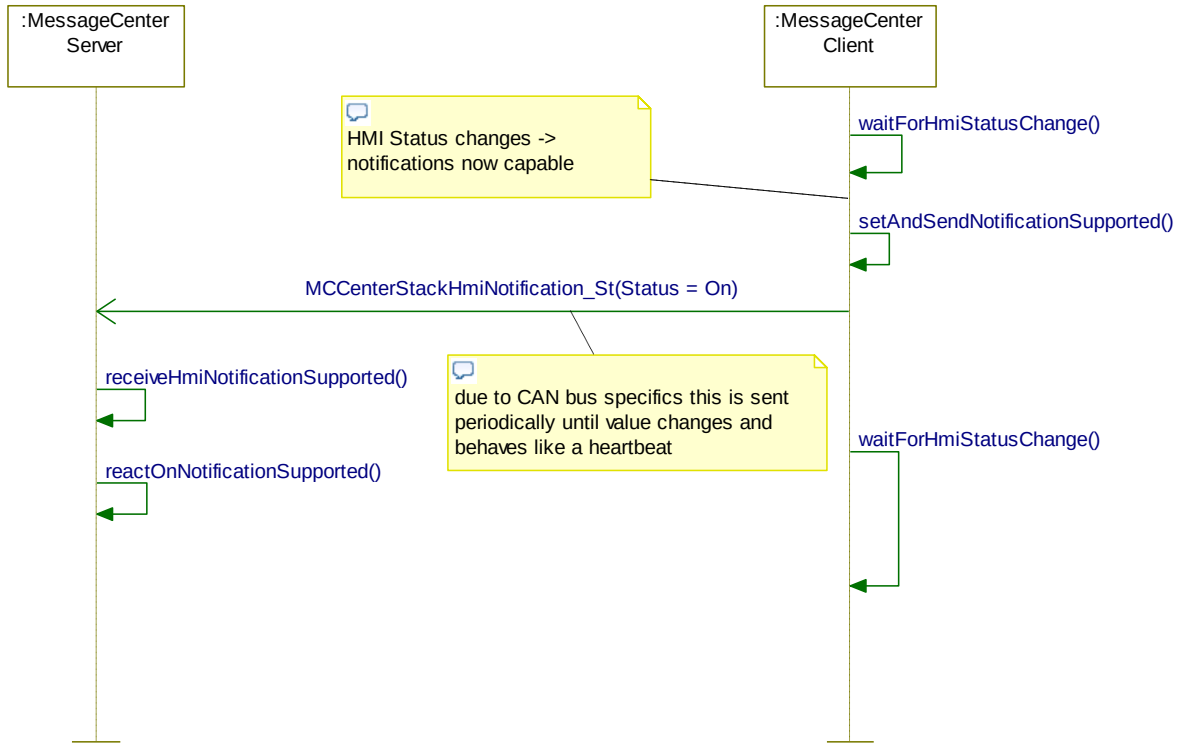
Post-condition

Client is waiting for change of HMI notification status.

Post-condition

CenterStackHmiNotification status signal is transmitted periodically using this value.

Sequence Diagram



4.8.4.3 MCv3-SD-REQ-362271/A-Broadcast Center Stack HMI Notification Capability changes to Off

Scenarios

Normal Usage

HMI notification status changes, since HMI is now not capable anymore to present notifications to the user. CenterStackHmiNotification status signal is updated with “Notification not capable” and transmitted periodically.

Constraints

Pre-condition

System is up and running.

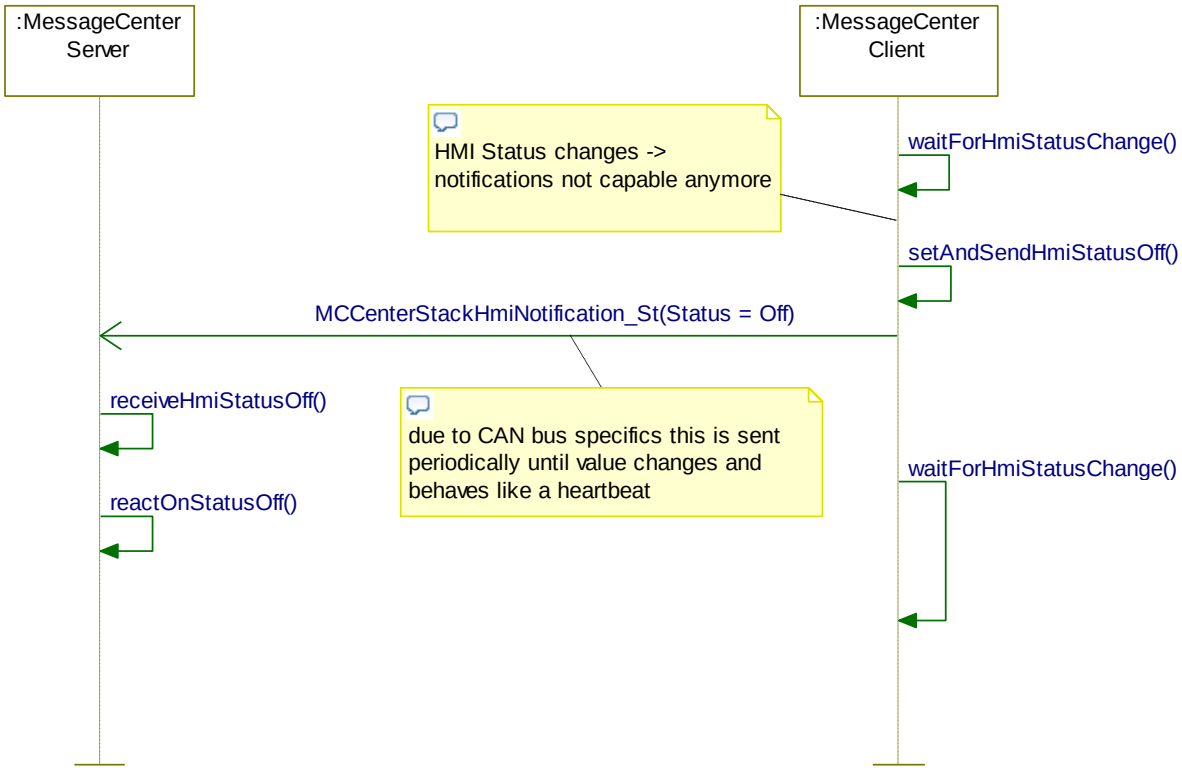
Post-condition

Client is waiting for change of HMI notification status.

Post-condition

CenterStackHmiNotification status signal is transmitted periodically using this value.

Sequence Diagram



5 APPENDIX A: Example of lookup table used in this spec

Please keep in mind following table is an example used in this spec to have it more demonstrative. For real table please refer to HMI documentation.

Lookup table examples (valid for MC Client and MC Server):

Msg ID (fictive)	HMI ID	HMI content	DD local	DD1-Type	DD2-Type	DD3-Type	Ok button	Choices	Comment
0	n/a	n/a	no	0 UnUsed	0 UnUsed	0 UnUsed	no	no	Deallocate all messages
1	n/a	n/a	no	0 UnUsed	0 UnUsed	0 UnUsed	no	no	Deallocate all messages except limited mode messages
2	n/a	n/a	no	0 UnUsed	0 UnUsed	0 UnUsed	no	no	Global Alert active (-> blank screen) - not removing from stack or deallocating
12	none	Travelling {xx.x} {unit} to POI	no	3 Float	4 Unit	0 UnUsed	no	no	MC Client must divide by 100 and remove least digit since only one after point will be shown DD3 and DD4 are not defined in layout -> don't care
158	W1087	Factory Keypad Code { X X X X X X X }	no	2 Integer2	2 Integer2	6 Digits	no	no	MC Client must interpret DD2 and DD1 as one integer DD3 carries 7 or 5 digits to show
211	W4110	Road Construction Ahead {XXXX} {unit}	yes	0 UnUsed	0 UnUsed	0 UnUsed	no	no	Since this is LHI dynamic data is already locally available, it will not be received via message center CAN signal
15	W100	Driver Door Ajar {OK}	no	0 UnUsed	0 UnUsed	0 UnUsed	yes	no	MC Server and MC Client know that an OK button for confirmation exists
124	W1800	Is Vehicle Plugged In ? {Yes} {No}	no	0 UnUsed	0 UnUsed	0 UnUsed	no	Choice1 = "Yes" Choice2 = "No"	MC Server and MC Client know that 2 Choices exist: Choice 1 = Yes, Choice 2 = No
73	W1653	Rear Park Aid Deactivated Trailer Attached	no	0 UnUsed	0 UnUsed	0 UnUsed	no	no	
45	W1805	Limited Performance Due to hot battery {OK}	no	0 UnUsed	0 UnUsed	0 UnUsed	yes	no	MC Server and MC Client know that an OK button for confirmation exists
28	W3486	Blind Spot Alert Deactivated Trailer Attached	no	0 UnUsed	0 UnUsed	0 UnUsed	yes	no	MC Server and MC Client know that an OK button for confirmation exists

		{OK}							
375	W3353a	Ready to Drive {XXX}% Charged	no	1 Integer	0 UnUsed	0 UnUsed	no	no	MC Server and MC Client know DD1 is an integer number
530	W4146	Return to {Drive Mode} ? Yes No	no	5 DriveMode	0 UnUsed	0 UnUsed	no	no	MC Server and MC Client know DD1 is an enumeration of drive modes

Msg ID: unique ID of table index
HMI ID: reference ID used in global message list
HMI content: schematic appearance how HMI might look
DD local: is dynamic data taken from local data, like e.g. in IODs.
DDx-Type: type of this dynamic data field; see APX-642758 as reference
OK button: does this message have an "OK" field?
Choices: does this message have any choices and which one is assigned to which content?
Comment: additional information on whole data set on this line and how to interpret to give additional understanding

6 APPENDIX B: Example table of dynamic data types used in this spec

Please keep in mind following table is an example used in this spec to have it more demonstrative. For real table please refer to HMI documentation.

Example Dynamic Data type list (valid for MC Client and MC Server):

Type	Name	Values	Comment
0	UnUsed	n/a	not applicable
1	Integer	0-65535	
2	Integer2	0-4294967295	Use 2 DDs as one Integer (=32Bit)
3	Float	0.00-655.35	multiply with 0.01
4	Unit	0x0 Km/h 0x1 mph 0x2 miles 0x3 feet	Enumeration for unit
5	DriveMode	0x00 SelDrvMde01 0x01 SelDrvMde02 0x02 SelDrvMde03 0x03 SelDrvMde04 0x04 SelDrvMde05 0x05 SelDrvMde06 0x06 SelDrvMde07 0x07 SelDrvMde08 0x08 SelDrvMde09 0x09 SelDrvMde10 0x0A SelDrvMde11 0x0B SelDrvMde12 0x0C SelDrvMde13 0x0D SelDrvMde14 0x0E SelDrvMde15 0x0F SelDrvMde16 0x10 SelDrvMde17 0x11 SelDrvMde18 0x12 SelDrvMde19 0x13 SelDrvMde20 0x14 SelDrvMde21 0x15 SelDrvMde22 0x16 SelDrvMde23 0x17 SelDrvMde24 0x18 SelDrvMde25 0x19 SelDrvMde26 0x1A SelDrvMde27 0x1B SelDrvMde28 0x1C SelDrvMde29 0x1D SelDrvMde30 0x1E SelDrvMde31 0x1F Faulty_default	Enumeration for drive mode Naming is different per carline, so only generic names are used here
6	Digits	0x0 7 digits 0x1 5 digits	Enumeration for keypad variant

Type: unique ID of table index
Name: term naming this type
Values: value range/enumeration values
Comment: additional information on whole data set on this line and how to interpret to give additional understanding