GENIVI Alliance

GENIVI Document IAmControlPlugin

Platform Compliance Specification

Version 2.0
Status: Published

Sponsored by:
GENIVI Alliance

Accepted for release by:
This document has not yet been accepted for release by the GENIVI Alliance Board of Directors.

Abstract:
This document is the GENIVI platform compliance specification.

Keywords: GENIVI

Table of contents

1 Introduction

This document describes general requirements on the platform, all external standards and all the components of the GENIVI platform release.

2 Change History

The following table shows the change history for this specification:

Date	Version	Author, Comment
6.12.11	1.0	Initial version, C Müller, BMW
2.3.12	1.1	Update for release
31.1.13	2.0	Update for foton release

3 Targeted Audience

Anyone wishing to apply for compliance should have this document at hand, as it complements the *GENIVI Platform Compliance Statement* document.

4 Categories

The compliance requirements within the GENIVI compliance program fall into the following general categories:

- *Standards*

The Standards list the compliance requirements that refer to standards external to GENIVI

- *Components from GENIVI component list*, that can be of the following compliance levels:
  - *PlaceholderComponent,* as described in GENIVI Compliance Explanation
  - *AbstractComponent,* as described in GENIVI Compliance Explanation
  - *SpecificComponent,* as described in GENIVI Compliance Explanation

5 Abbreviations

In further text the following abbreviations (ABB) will be used for forming the IDs:

- ST for Standards
- PC for PlaceholderComponent
- AC for AbstractComponent
- SC for SpecificComponent

6 Format of the entries in this document

The entries pertaining to components have the following format:

Field	*Description*
ID	unique ID, text formed as: <ABB>-<auto incrementing number>.<sub code>
Name	name with optional link (URL) to the external reference (e.g. The project home page)
Compliance Level	as defined in *Categories*.
Priority	as described in GENIVI Compliance Explanation
Implementation constraints	Contains the version(s) allowed May contain compile and runtime configurations
Description	textual clarification
Requirements	For PC, AC contains a table of requirements

7 Standards

GENIVI Compliance Release 1.0 does not require references to external standards.

8 Component Overview
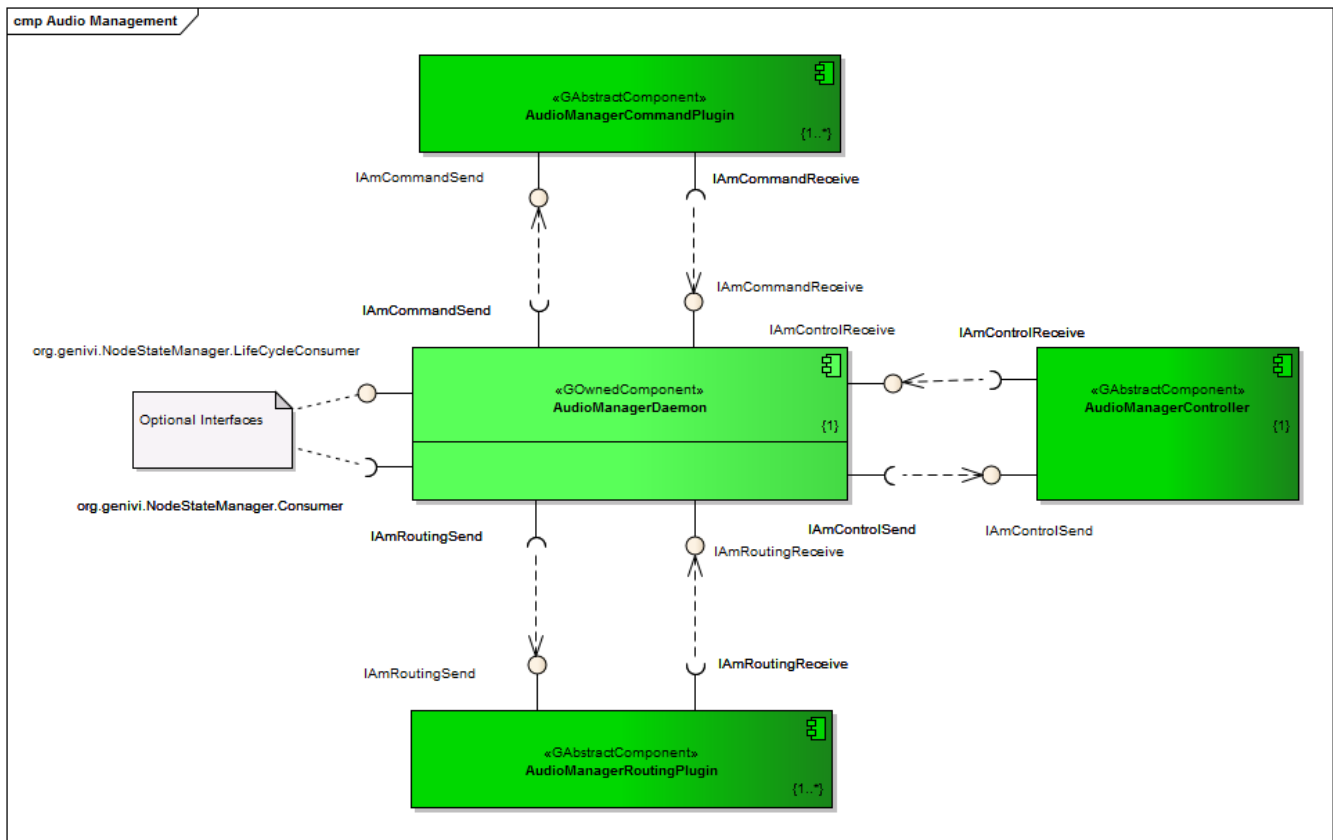
9 Component Interaction

9.1 Toolbox

Type: **Package**
Status: Reworked. Version 1.0. Phase 1.0.
Package: Control Interface
Detail: *Created on 24.08.2011. Last modified on 10.10.2011*
GUID: {381F5748-B9F8-4e8a-B955-62222423FA69}

This packet holds all actions that can be used by the controller. These tasks are at disposal for the AudioManagerController to fullfill his tasks.

tool_AsyncSetSinkNotificationConfiguration
Type: **Package**
Status: Proposed. Version 1.0. Phase 1.0.
Package: Toolbox
Detail: *Created on 29.01.2013. Last modified on 29.01.2013*
GUID: {FCA04F87-958B-4012-8E18-61AF59B86FBE}

tool_AsyncSetSinkNotificationConfiguration - *(Interaction diagram)*

Created By: q232968 *on* 29.01.2013
Last Modified: 29.01.2013
Version: 1.0. *Locked:* False
GUID: {875ADB5B-2619-4522-89AD-6D4AD0628659}

Figure: 1

actions_AsynchronousSetSinkNotificationConfiguration

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSinkNotificationConfiguration *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{B3CEBF1D-F62F-4b91-868A-DB1B7328F151}

Connector	Source	Target	Notes
Sequence Source -> Destination	Public IAmControlReceive	Public actions_AsynchronousSetSinkNotificationConfiguration	
Sequence Source -> Destination	Public actions_AsynchronousSetSinkNotificationConfiguration	Public IAmControlReceive	
Sequence Source -> Destination	Public actions_AsynchronousSetSinkNotificationConfiguration	Public IAmControlReceive	

change

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSinkNotificationConfiguration *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{4A80F484-9DD2-4dab-8951-B3EF2E4EEA58}

sink

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSinkNotificationConfiguration *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{3615F69D-78AB-480c-8D82-CF04ED709CCF}

sinkNotificationConfiguration

Type:	**InteractionFragment**

Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSinkNotificationConfiguration *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{DF6A53DA-52BB-41a9-B956-A9BA6BD86878}

tool_AsyncSetSourceNotificationConfiguration

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 31.01.2013. Last modified on 31.01.2013*
GUID:	{6F0CB6F4-180A-4153-AE1B-BBE54A5B3B2F}

tool_AsyncSetSourceNotificationConfiguration - *(Interaction diagram)*

Created By:	q232968 *on* 29.01.2013
Last Modified:	31.01.2013
Version:	1.0. *Locked:* False
GUID:	{F89B0574-5E95-4beb-9D28-970813A4C112}
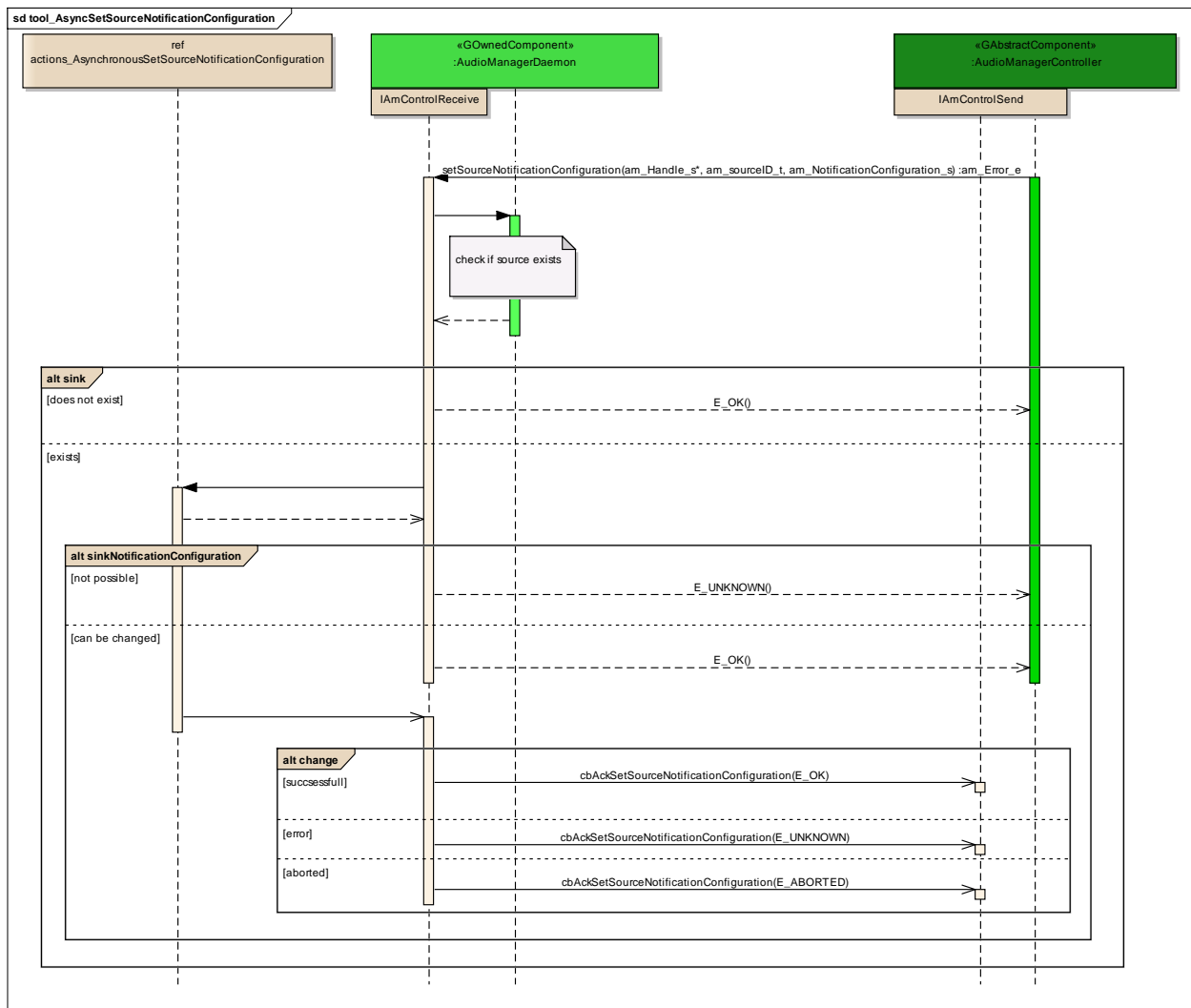
Figure: 2

actions_AsynchronousSetSourceNotificationConfiguration
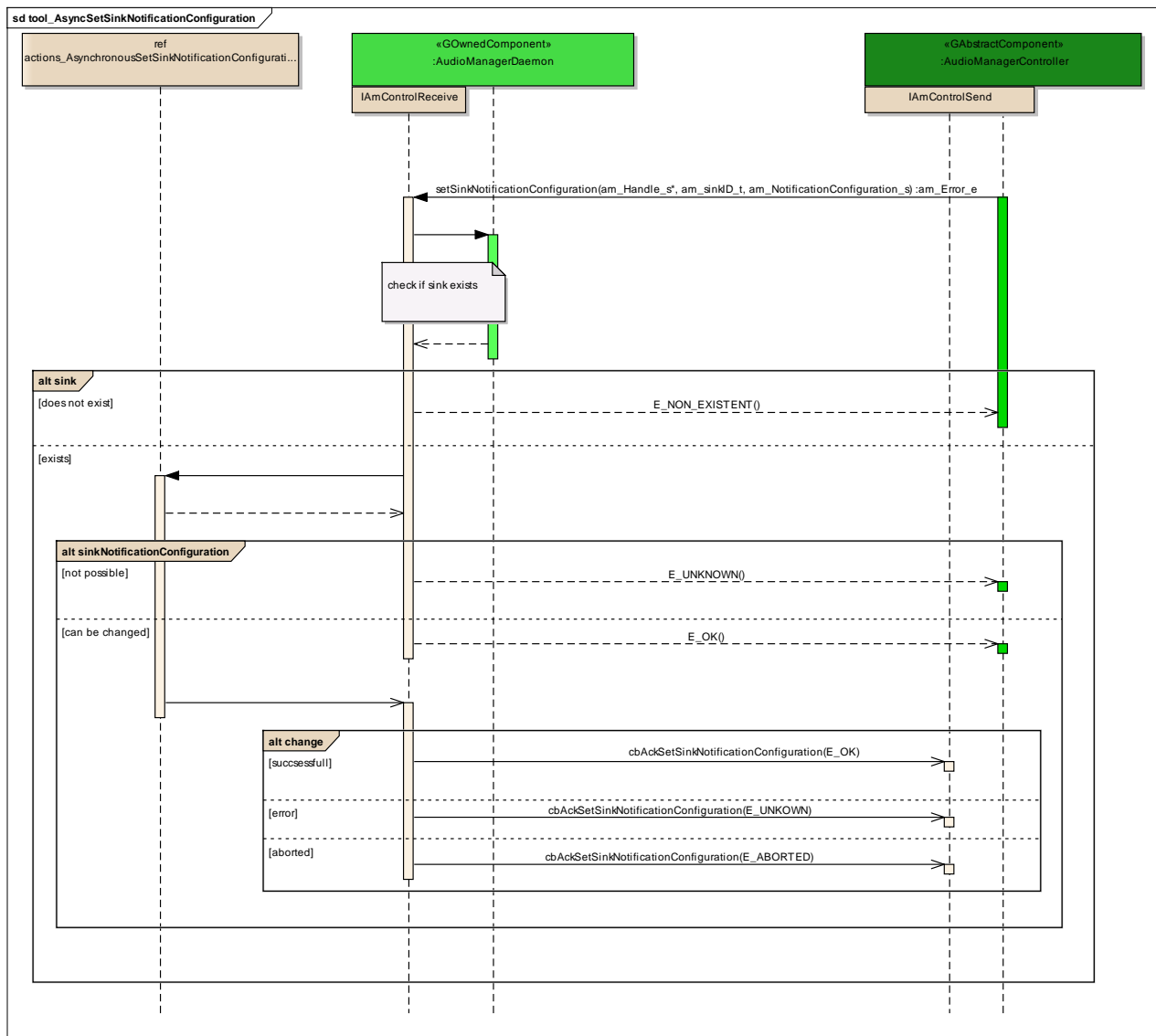
Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: tool_AsyncSetSourceNotificationConfiguration *Keywords:*
Detail: *Created on 31.01.2013. Last modified on 31.01.2013.*
GUID: {1EDC895D-012A-497d-827B-C3297392FC3A}

Connections

Connector	Source	Target	Notes

Connector	Source	Target	Notes
Sequence Source -> Destination	Public IAmControlReceive	Public actions_Asynchron ousSetSourceNotifi cationConfiguration	
Sequence Source -> Destination	Public actions_Asynchron ousSetSourceNotifi cationConfiguration	Public IAmControlReceive	
Sequence Source -> Destination	Public actions_Asynchron ousSetSourceNotifi cationConfiguration	Public IAmControlReceive	

change

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceNotificationConfiguration *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{89BEE17B-6C9D-40a4-9609-EC313D5D88B3}

sink

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceNotificationConfiguration *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{2F079AA3-0817-4360-B3B1-A6EC9EEC3011}

sinkNotificationConfiguration

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.

Package:	tool_AsyncSetSourceNotificationConfiguration *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{5B05FDE1-5562-4754-9F60-FB844E85B665}

tool_SendMainSinkNotification

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013*
GUID:	{9FECEEF8-C3CA-44c3-9452-FFCA2D87341C}

tool_SendMainSinkNotification - *(Interaction diagram)*

Created By:	q232968 *on* 29.01.2013
Last Modified:	29.01.2013
Version:	1.0. *Locked:* False
GUID:	{5F51F89F-8C74-4a5f-B66C-2C503EF5EBB5}
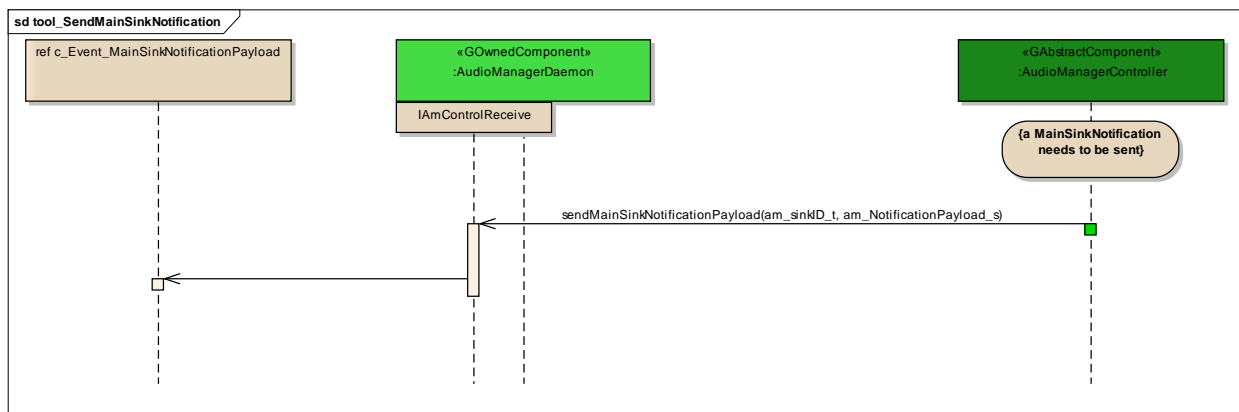
Figure: 3

c_Event_MainSinkNotificationPayload

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_SendMainSinkNotification *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{87DE524D-8DB2-431b-87D6-3E5619AD108C}

Connector	Source	Target	Notes
Sequence Source -> Destination	Public IAmControlReceive	Public c_Event_MainSink NotificationPayload	

tool_sendMainSourceNotification

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013*
GUID:	{494C2AC6-F326-49bc-82A7-7B63DA860DDC}

tool_sendMainSourceNotification - *(Interaction diagram)*

Created By:	q232968 *on* 29.01.2013
Last Modified:	29.01.2013
Version:	1.0. *Locked:* False
GUID:	{9DA1D111-1743-46a8-A9A3-1FFD72B81128}
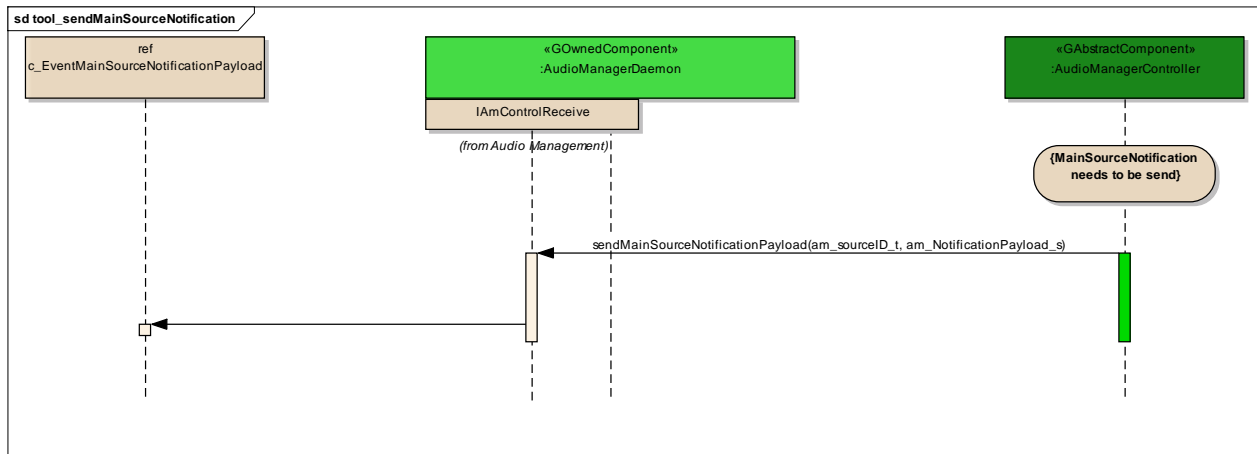
Figure: 4

c_EventMainSourceNotificationPayload

Type:	**InteractionOccurrence**

Connector	Source	Target	Notes
Sequence Source -> Destination	Public IAmControlReceive	Public c_EventMainSourceNotificationPayload	

tool_AsyncMessaging

tool_AsyncMessaging - *(Interaction diagram)*

The sequence shown here is the same for all asynchronous transmitted messages.
It shows how the handles are created and destroyed and what the daemon does to evaluate the correct routing plugin
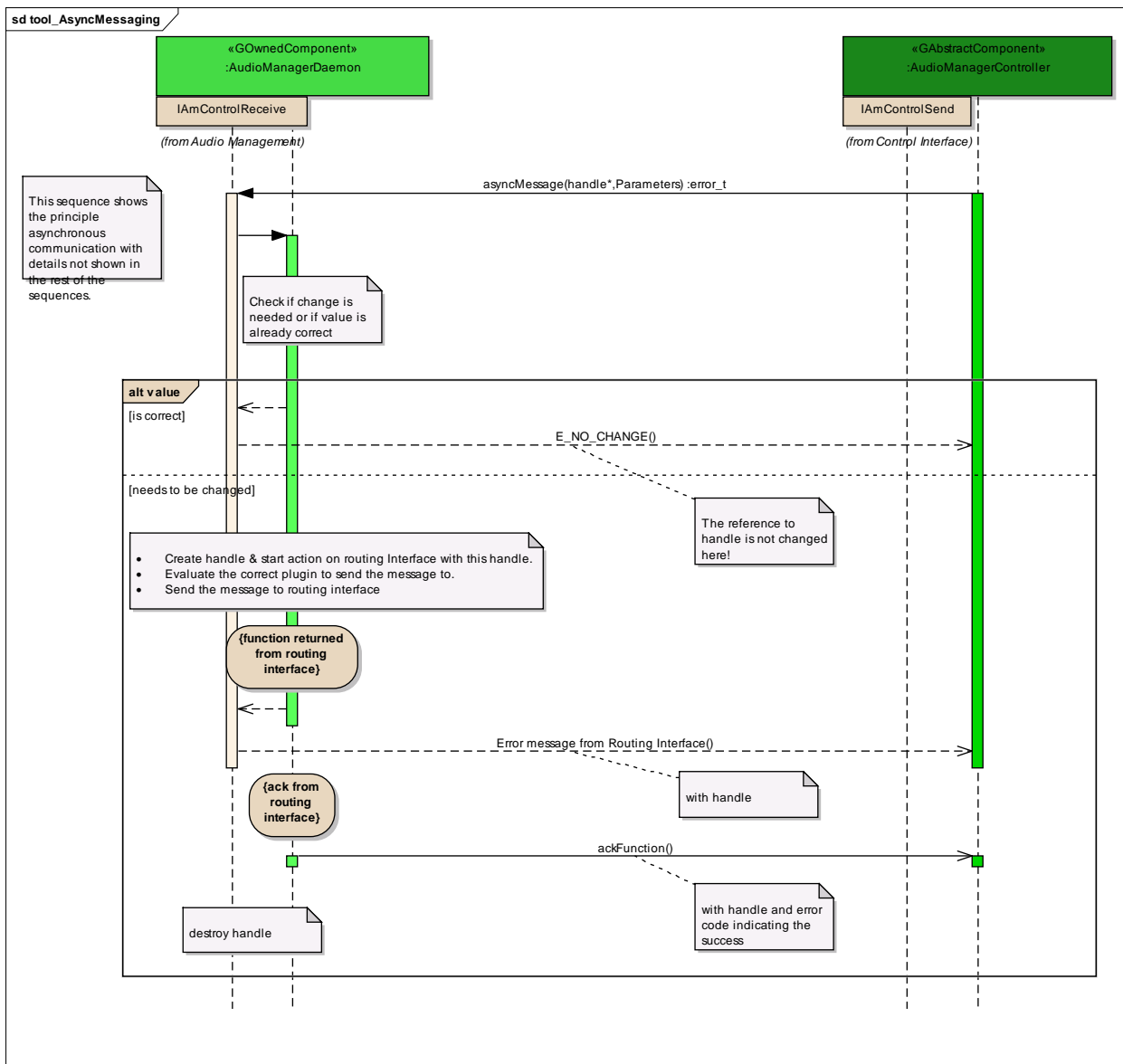
Figure: 5

value

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncMessaging *Keywords:*
Detail:	*Created on 30.09.2011. Last modified on 30.09.2011.*
GUID:	{DEB1D05C-0AB5-418a-99B8-B060662AFEB1}

tool_AsyncSetSourceState

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 08.09.2011. Last modified on 11.10.2011*
GUID:	{049AF3A5-E4E8-4057-80D7-E58DBC3C8C5B}

tool_AsyncSetSourceState - *(Interaction diagram)*

Created By:	christian *on* 08.09.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{53A064EE-26B2-4e0d-9604-8B6FFC0FAE31}
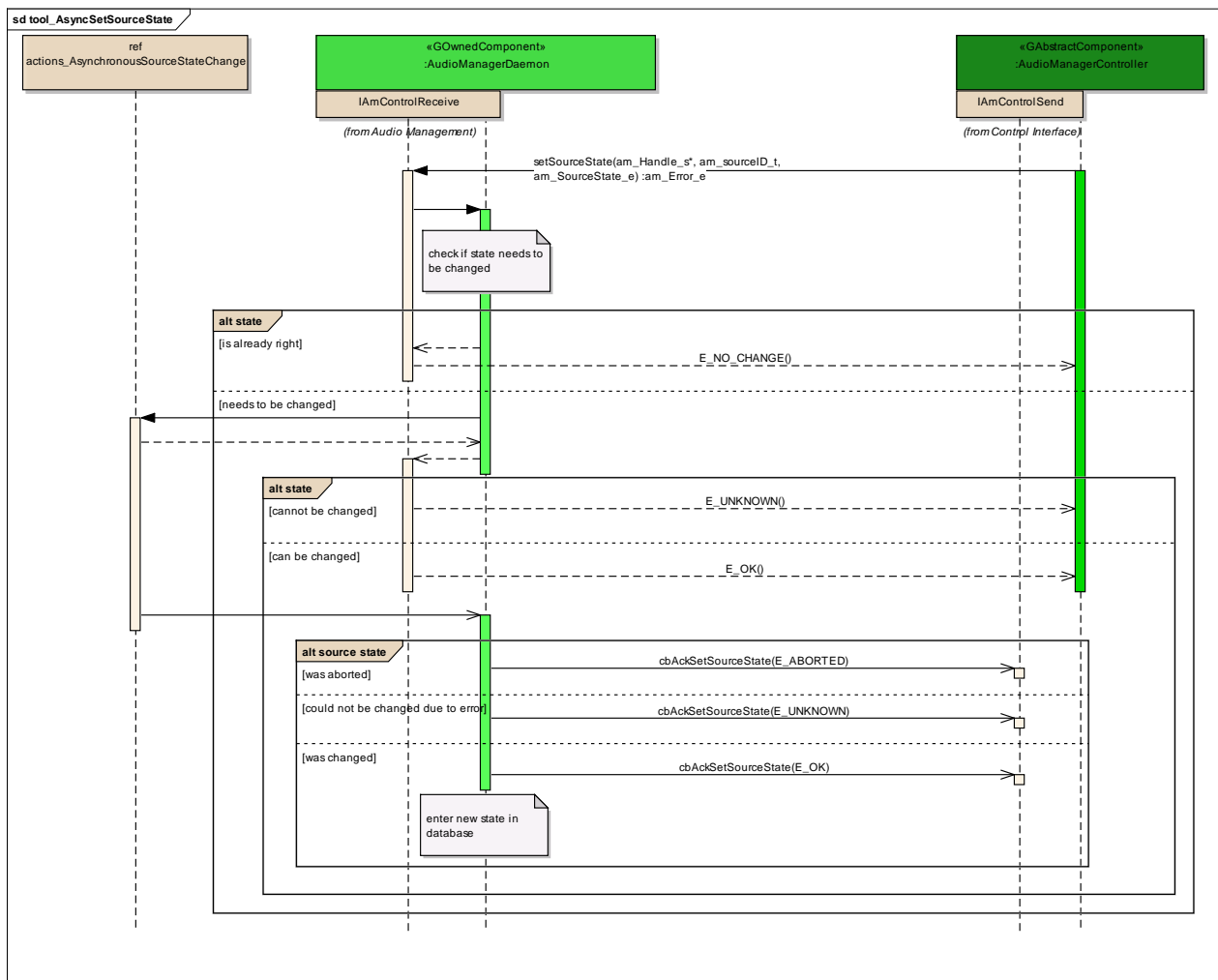
Figure: 6

actions_AsynchronousSourceStateChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceState *Keywords:*
Detail:	Created on 08.09.2011. Last modified on 11.10.2011.
GUID:	{A64B4922-629E-4c61-89F7-AF02F0B56AB2}

Connections

Connector	Source	Target	Notes
Sequence Source -> Destination	Public <anonymous>	Public actions_AsynchronousSourceStateChange	
Sequence Source -> Destination	Public actions_AsynchronousSourceStateChange	Public <anonymous>	
Sequence Source -> Destination	Public actions_AsynchronousSourceStateChange	Public <anonymous>	

source state

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceState *Keywords:*
Detail:	Created on 30.09.2011. Last modified on 30.09.2011.
GUID:	{C9416652-8662-4d8a-A67C-F254768BD81C}

state

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceState *Keywords:*
Detail:	*Created on 08.09.2011. Last modified on 08.09.2011.*
GUID:	{B3471CAB-BC2F-400d-981E-A426BAD3D1C4}

state

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceState *Keywords:*
Detail:	*Created on 08.09.2011. Last modified on 08.09.2011.*
GUID:	{A51AD54D-6D81-4cab-85F5-4A37AAC72863}

tool_AsyncSetSourceVolume

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 05.09.2011. Last modified on 11.10.2011*
GUID:	{3887BA01-B6BB-4d22-A129-E0D3BECCC358}

tool_AsyncSetSourceVolume - *(Interaction diagram)*

Created By:	christian *on* 05.09.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{4B8A8FAD-8734-4682-81D3-5CC013468D43}
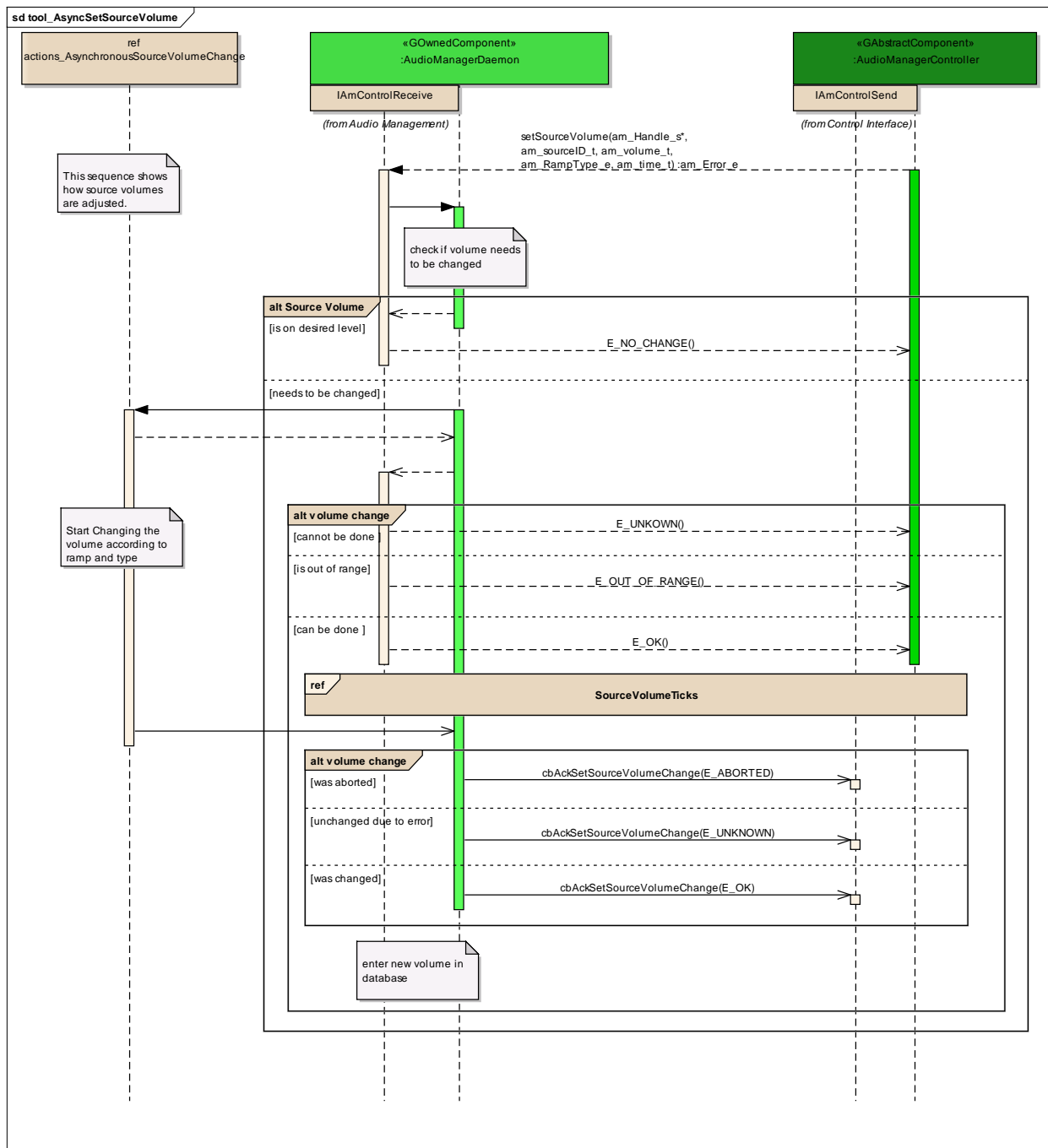
Figure: 7

Source Volume

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceVolume *Keywords:*
Detail:	*Created on 08.09.2011. Last modified on 08.09.2011.*

GUID:	{BA816171-81EC-4dac-B456-AFFD76D1245D}

SourceVolumeTicks

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceVolume *Keywords:*
Detail:	*Created on 04.10.2011. Last modified on 04.10.2011.*
GUID:	{809A42C5-6F57-4964-9A66-3D1C6A967103}

actions_AsynchronousSourceVolumeChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceVolume *Keywords:*
Detail:	*Created on 05.09.2011. Last modified on 11.10.2011.*
GUID:	{617863F3-4D31-4bf9-B301-3E44269FBD6B}

Connections

Connector	Source	Target	Notes
Sequence Source -> Destination	Public <anonymous>	Public actions_Asynchron ousSourceVolumeC hange	
Sequence Source -> Destination	Public actions_Asynchron ousSourceVolumeC hange	Public <anonymous>	
Sequence Source -> Destination	Public actions_Asynchron ousSourceVolumeC hange	Public <anonymous>	

Connector	Source	Target	Notes

volume change

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceVolume *Keywords:*
Detail:	*Created on 04.10.2011. Last modified on 04.10.2011.*
GUID:	{B59B6D47-79DD-4628-9F11-C78ABA6C83F9}

volume change

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceVolume *Keywords:*
Detail:	*Created on 05.09.2011. Last modified on 05.09.2011.*
GUID:	{7ACDC340-3461-4845-BEDC-AFCFFC6A0782}

tool_AsyncSetSinkVolume

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 06.09.2011. Last modified on 12.04.2012*
GUID:	{FDE8B250-96A8-476c-80AD-DBF77354CF90}

tool_AsyncSetSinkVolume - *(Interaction diagram)*

Created By:	christian *on* 06.09.2011
Last Modified:	29.11.2012
Version:	1.0. *Locked:* False
GUID:	{B26BD11E-514E-46ec-81E5-9D70CE165FD9}
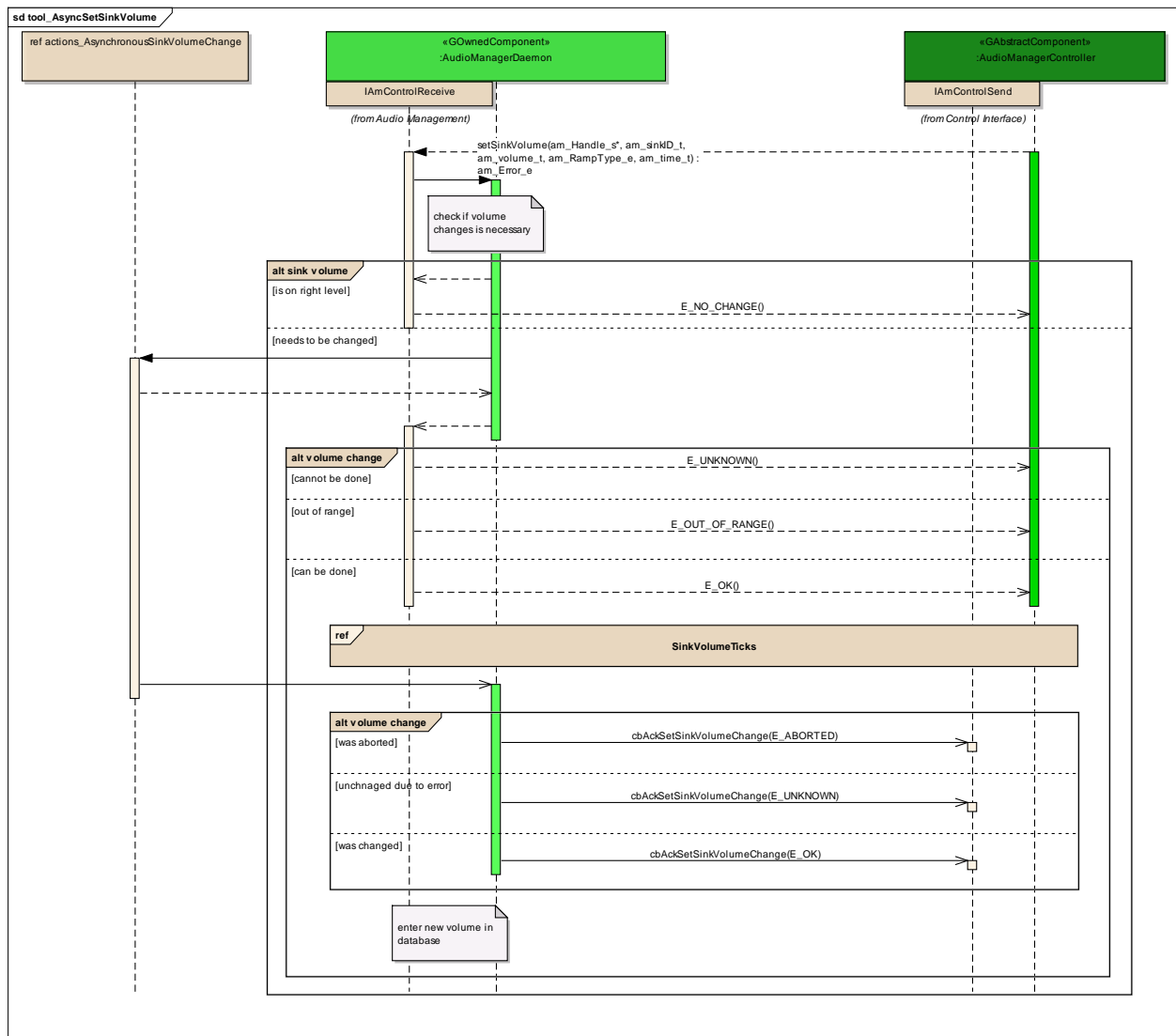
Figure: 8

SinkVolumeTicks

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSinkVolume *Keywords:*
Detail:	*Created on 04.10.2011. Last modified on 04.10.2011.*
GUID:	{E06AB8EA-4F29-4d21-8540-B80FFFBE7C19}

actions_AsynchronousSinkVolumeChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSinkVolume *Keywords:*
Detail:	*Created on 06.09.2011. Last modified on 11.10.2011.*
GUID:	{930941AE-1CC3-47b4-97DF-61B225E3C1A2}

Connections

Connector	Source	Target	Notes
Sequence Source -> Destination	Public <anonymous>	Public actions_AsynchronousSinkVolumeChange	
Sequence Source -> Destination	Public actions_AsynchronousSinkVolumeChange	Public <anonymous>	
Sequence Source -> Destination	Public actions_AsynchronousSinkVolumeChange	Public <anonymous>	

sink volume

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSinkVolume *Keywords:*
Detail:	*Created on 08.09.2011. Last modified on 04.10.2011.*
GUID:	{5F5971A5-1D23-4622-843D-7C7AF309D25C}

volume change

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.

Package:	tool_AsyncSetSinkVolume	*Keywords:*
Detail:	*Created on 06.09.2011. Last modified on 06.09.2011.*	
GUID:	{F21A0DBA-0680-4bc4-8964-71C9150D8426}	

volume change

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSinkVolume *Keywords:*
Detail:	*Created on 04.10.2011. Last modified on 04.10.2011.*
GUID:	{C13CF26D-E278-48d1-9828-3A728A2A6672}

tool_AsyncStopAction

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 08.09.2011. Last modified on 11.10.2011*
GUID:	{A6C0F29B-5F9B-4c6a-BE4B-92B401761716}

tool_AsyncStopAction - *(Interaction diagram)*

Created By:	christian *on* 08.09.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{2AE56F06-C73A-4899-B5A7-F5F514878838}
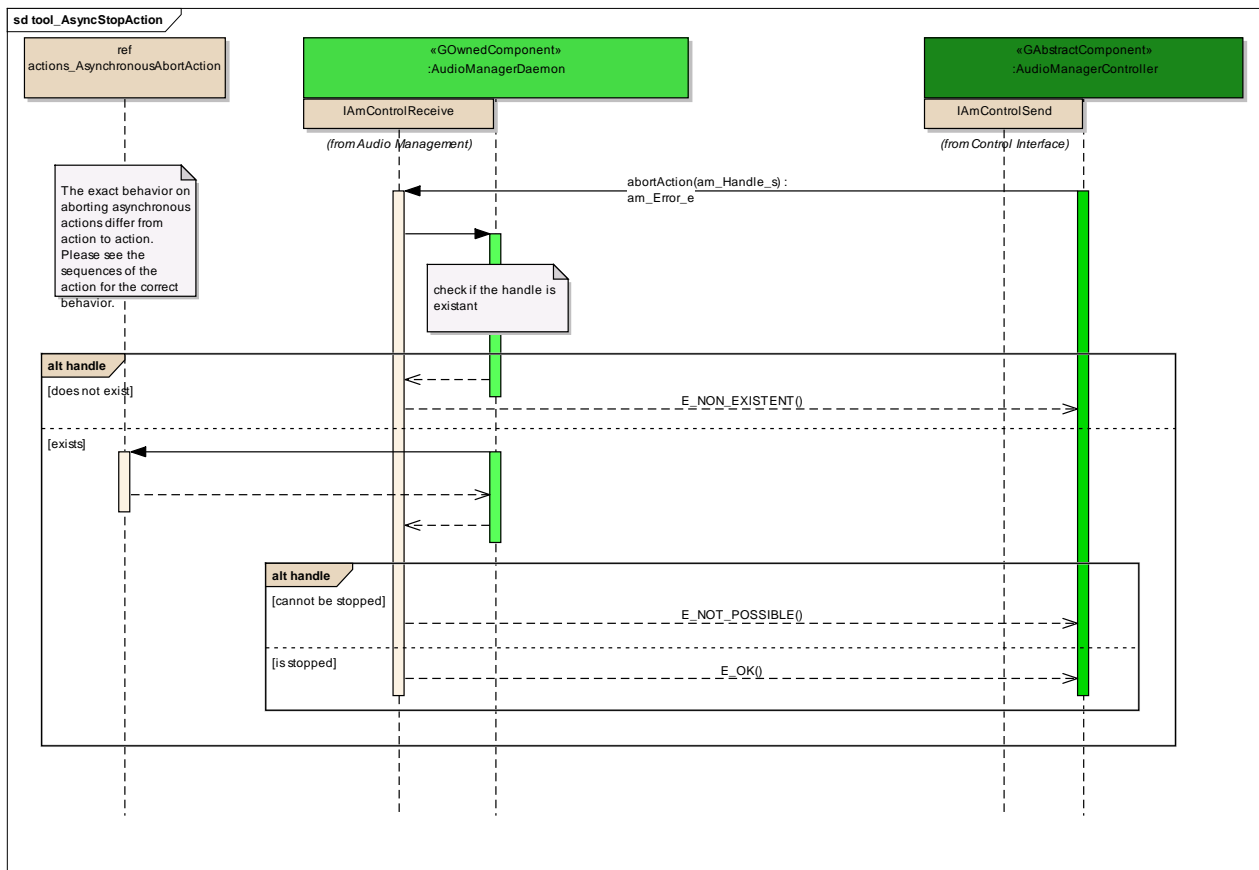
Figure: 9

actions_AsynchronousAbortAction

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: tool_AsyncStopAction   *Keywords:*
Detail: Created on 08.09.2011. Last modified on 11.10.2011.
GUID: {29BF4CA5-10EA-4fd0-AE78-5358D2C9E9A9}

Connections

Connector	Source	Target	Notes
Sequence Source -> Destination	Public <anonymous>	Public actions_Asynchron ousAbortAction	
Sequence Source -> Destination	Public	Public	

Connector	Source	Target	Notes
	actions_Asynchron ousAbortAction	<anonymous>	

handle

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncStopAction *Keywords:*
Detail:	*Created on 08.09.2011. Last modified on 08.09.2011.*
GUID:	{2CC3C37E-93D4-4e87-932A-44606D3C309D}

handle

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncStopAction *Keywords:*
Detail:	*Created on 08.09.2011. Last modified on 08.09.2011.*
GUID:	{C128FDCF-687C-4a4a-B75D-E56828835FAD}

tool_AsyncConnection

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 24.08.2011. Last modified on 11.10.2011*
GUID:	{5DC2BDF9-DB07-4419-9974-C4205EBF69D5}

tool_AsyncConnection - *(Interaction diagram)*

Created By:	christian *on* 24.08.2011
Last Modified:	01.12.2011
Version:	1.0. *Locked:* False
GUID:	{C381D56A-FBB4-45fd-B7D9-54713CD34CF8}
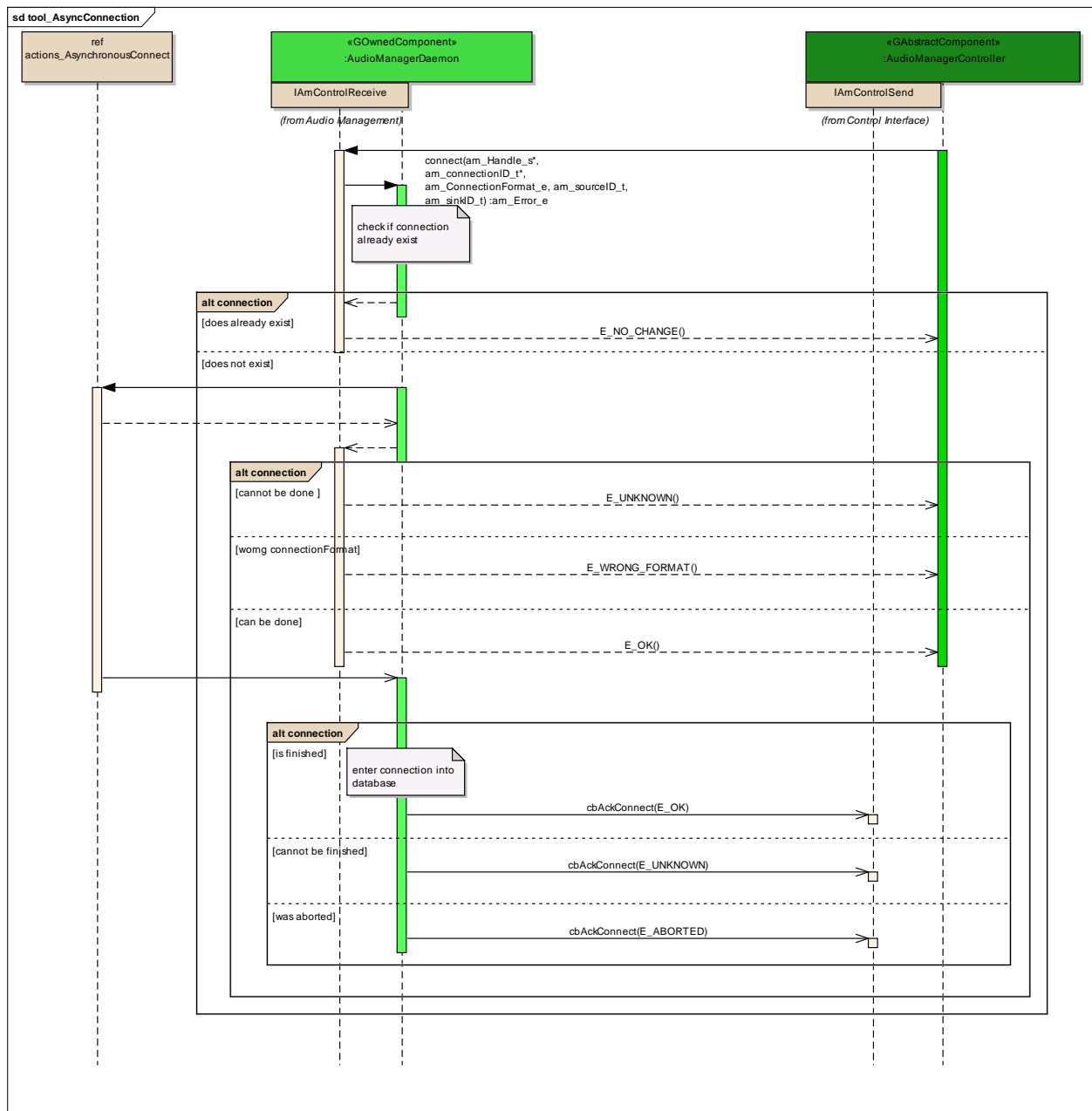
Figure: 10

actions_AsynchronousConnect

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncConnection *Keywords:*
Detail:	*Created on 24.08.2011. Last modified on 11.10.2011.*
GUID:	{9EE59FB3-71ED-4f79-9279-C8A24721B91C}

Connector	Source	Target	Notes
Sequence Source -> Destination	Public <anonymous>	Public actions_Asynchron ousConnect	
Sequence Source -> Destination	Public actions_Asynchron ousConnect	Public <anonymous>	
Sequence Source -> Destination	Public actions_Asynchron ousConnect	Public <anonymous>	

connection

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncConnection *Keywords:*
Detail:	Created on 08.09.2011. Last modified on 08.09.2011.
GUID:	{E486032E-E24B-465b-A47E-45C37138361B}

connection

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncConnection *Keywords:*
Detail:	Created on 24.08.2011. Last modified on 24.08.2011.
GUID:	{963A12A0-E4D9-490a-8389-C3C6CFBE4A4C}

connection

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncConnection *Keywords:*
Detail:	*Created on 08.09.2011. Last modified on 08.09.2011.*
GUID:	{6B1F3D91-D96E-49da-AA0F-3C5D1D2FA652}

tool_AsyncDisconnection

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 24.08.2011. Last modified on 11.10.2011*
GUID:	{A9831417-AF48-40f3-B5DD-5EDC4238976C}

tool_AsyncDisconnection - *(Interaction diagram)*

Created By:	christian *on* 24.08.2011
Last Modified:	28.12.2011
Version:	1.0. *Locked:* False
GUID:	{4FEC2CD1-A04C-4a85-A672-DEE4FB9CEA04}
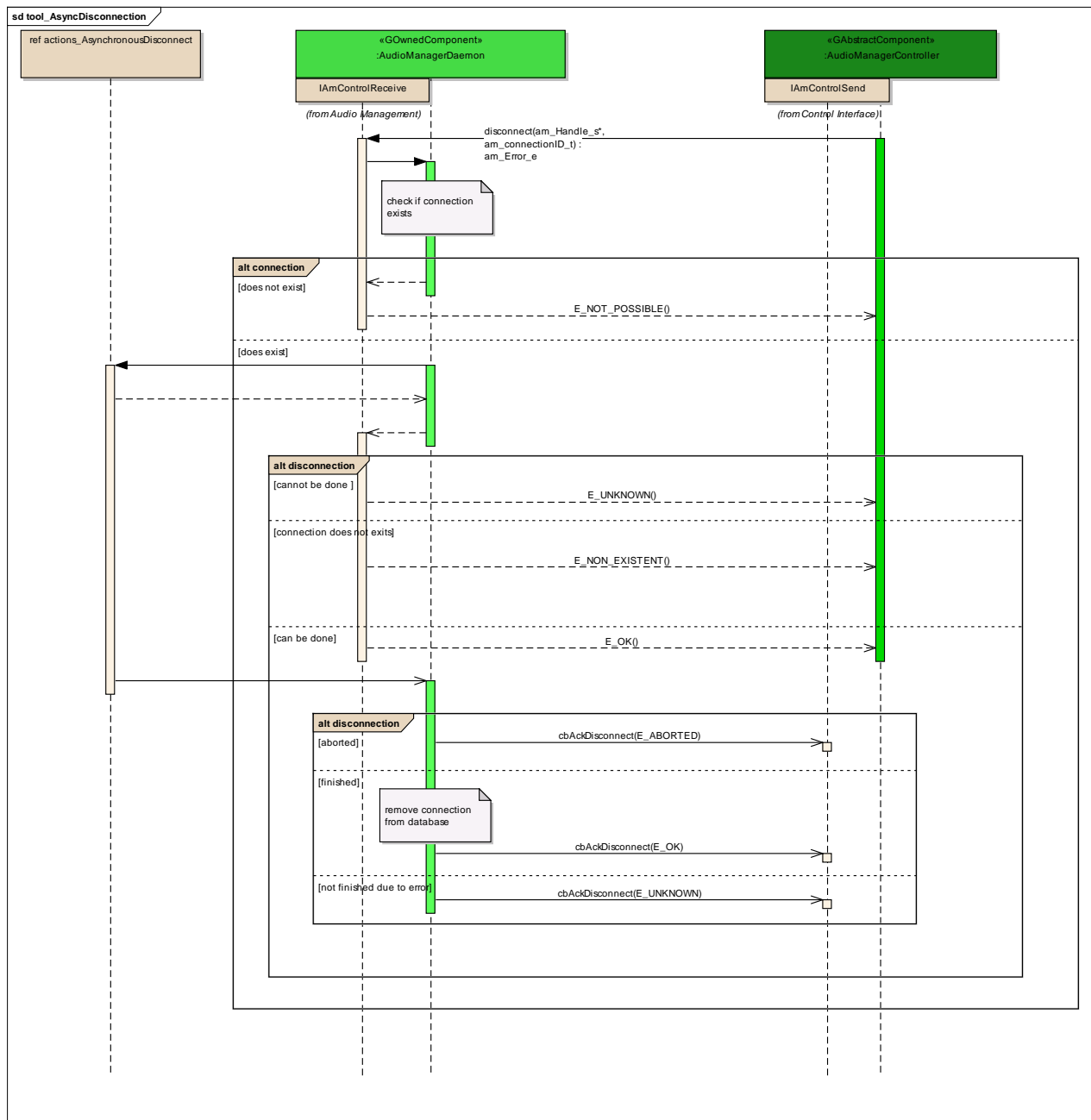
Figure: 11

actions_AsynchronousDisconnect

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncDisconnection *Keywords:*
Detail:	Created on 24.08.2011. Last modified on 11.10.2011.
GUID:	{CA595642-5385-4c1d-B173-29F0DA3F577F}

Connector	Source	Target	Notes
Sequence Source -> Destination	Public <anonymous>	Public actions_Asynchron ousDisconnect	
Sequence Source -> Destination	Public actions_Asynchron ousDisconnect	Public <anonymous>	
Sequence Source -> Destination	Public actions_Asynchron ousDisconnect	Public <anonymous>	

connection

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncDisconnection *Keywords:*
Detail:	*Created on 05.10.2011. Last modified on 05.10.2011.*
GUID:	{C8E88406-8F83-4e47-9860-935592B13041}

disconnection

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncDisconnection *Keywords:*
Detail:	*Created on 05.10.2011. Last modified on 05.10.2011.*
GUID:	{620ECFE3-3AA1-44a1-8F7E-5BBF8FE4951F}

disconnection

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncDisconnection *Keywords:*
Detail:	*Created on 24.08.2011. Last modified on 24.08.2011.*
GUID:	{409D4597-B366-4378-8604-E090944FC858}

tool_AsyncSetSourceSoundProperty

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 06.09.2011. Last modified on 11.10.2011*
GUID:	{635AB6EB-A89F-4380-948B-0E9B93C273E8}

tool_AsyncSetSourceSoundProperty - *(Interaction diagram)*

Created By:	christian *on* 06.09.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{16C71D1C-80D3-447b-9053-B6046D2591E6}
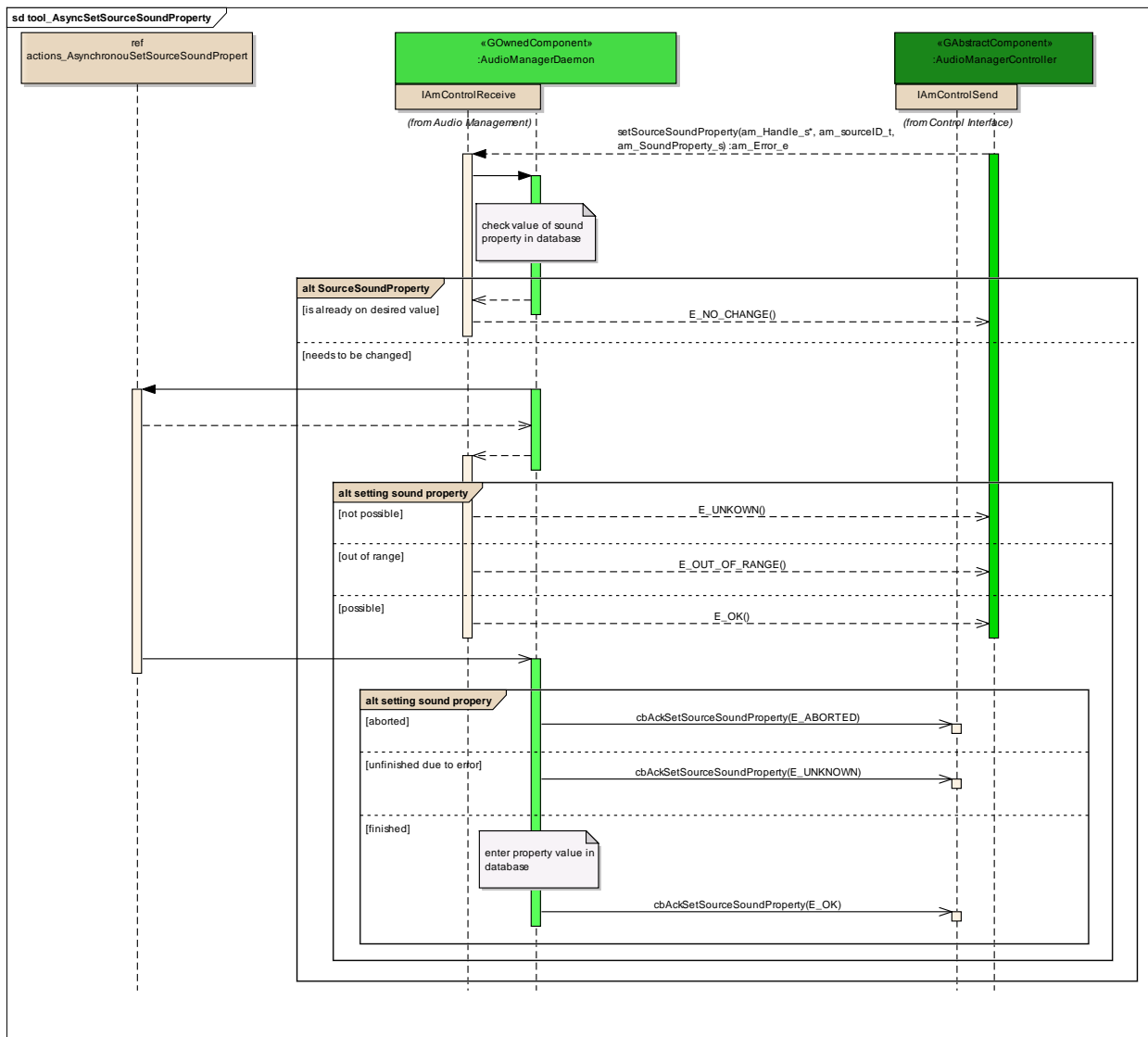
Figure: 12

SourceSoundProperty

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceSoundProperty *Keywords:*
Detail:	*Created on 08.09.2011. Last modified on 08.09.2011.*
GUID:	{6B4045B4-D75E-4612-BD9F-8A6A9883467C}

actions_AsynchronouSetSourceSoundPropert

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceSoundProperty *Keywords:*
Detail:	*Created on 06.09.2011. Last modified on 11.10.2011.*
GUID:	{55EB9DCE-96F9-4de1-B035-AD492A30080C}

Connections

Connector	Source	Target	Notes
Sequence Source -> Destination	Public <anonymous>	Public actions_Asynchron ouSetSourceSound Propert	
Sequence Source -> Destination	Public actions_Asynchron ouSetSourceSound Propert	Public <anonymous>	
Sequence Source -> Destination	Public actions_Asynchron ouSetSourceSound Propert	Public <anonymous>	

setting sound property

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncSetSourceSoundProperty *Keywords:*
Detail:	*Created on 06.09.2011. Last modified on 06.09.2011.*
GUID:	{35C01917-4666-4080-9C1A-E77DA0EB776C}

setting sound propery

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.

Package:	tool_AsyncSetSourceSoundProperty	Keywords:
Detail:	*Created on 05.10.2011. Last modified on 05.10.2011.*
GUID:	{AECD44AD-857D-446b-B110-607842CF472E}

tool_AsyncSetSinkSoundProperty

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 15.09.2011. Last modified on 11.10.2011*
GUID:	{B46AB621-8262-4572-A361-DD52B5277A91}

tool_AsyncSetSinkSoundProperty - *(Interaction diagram)*

Created By:	christian *on* 15.09.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{D03A41B1-B24D-49aa-9EFF-B87FBE4E052B}
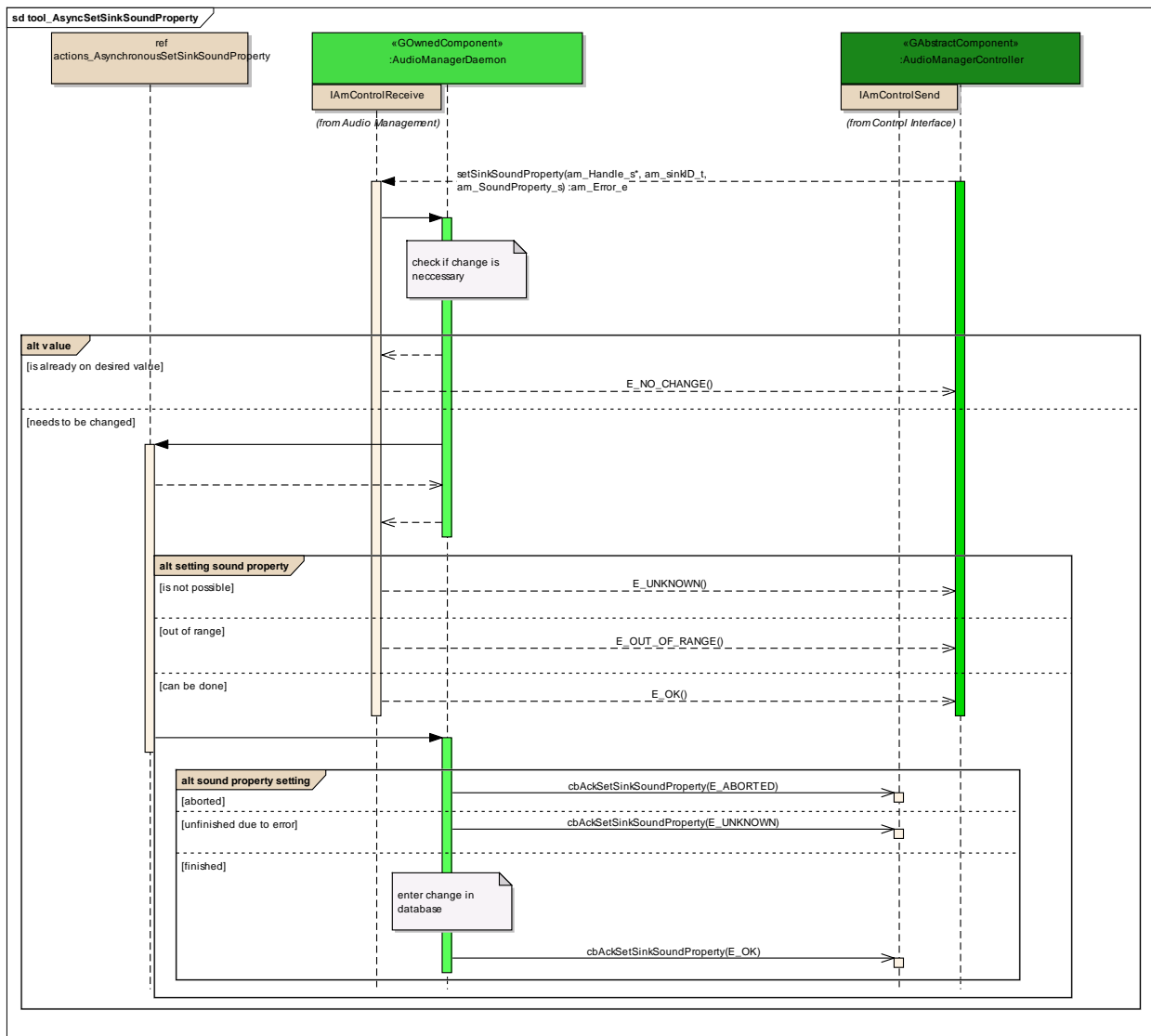
Figure: 13

actions_AsynchronousSetSinkSoundProperty

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: tool_AsyncSetSinkSoundProperty   *Keywords:*
Detail: *Created on 22.09.2011. Last modified on 11.10.2011.*
GUID: {5700B8B7-2134-4c07-87AC-97D632D95620}

Connections

Connector	Source	Target	Notes
Sequence Source -> Destination	Public <anonymous>	Public actions_Asynchron ousSetSinkSoundPr operty	
Sequence Source -> Destination	Public actions_Asynchron ousSetSinkSoundPr operty	Public <anonymous>	
Sequence Source -> Destination	Public actions_Asynchron ousSetSinkSoundPr operty	Public <anonymous>	

setting sound property

Type: **InteractionFragment**
Status: Proposed. Version 1.0. Phase 1.0.
Package: tool_AsyncSetSinkSoundProperty   *Keywords:*
Detail: *Created on 22.09.2011. Last modified on 22.09.2011.*
GUID: {BE8BE259-A14E-48d7-934B-FAD11614F1C1}

sound property setting

Type: **InteractionFragment**
Status: Proposed. Version 1.0. Phase 1.0.
Package: tool_AsyncSetSinkSoundProperty   *Keywords:*
Detail: *Created on 07.10.2011. Last modified on 07.10.2011.*
GUID: {7B2E5CC6-B59A-442e-B875-0E7AE2F116A2}

value

Type: **InteractionFragment**
Status: Proposed. Version 1.0. Phase 1.0.

Package:	tool_AsyncSetSinkSoundProperty	*Keywords:*	
Detail:	*Created on 22.09.2011. Last modified on 22.09.2011.*		
GUID:	{772BD1C6-E4AF-40dc-B5F4-759C13E936A8}		

tool_AsyncCrossFade

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 07.09.2011. Last modified on 12.10.2011*
GUID:	{6DE9E3B2-1F7D-4f5c-8B02-F1FF395C20F5}

tool_AsyncCrossFade - *(Interaction diagram)*

Created By:	christian *on* 07.09.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{7B2654FD-485F-41c6-B5DE-C9D4BAA68983}
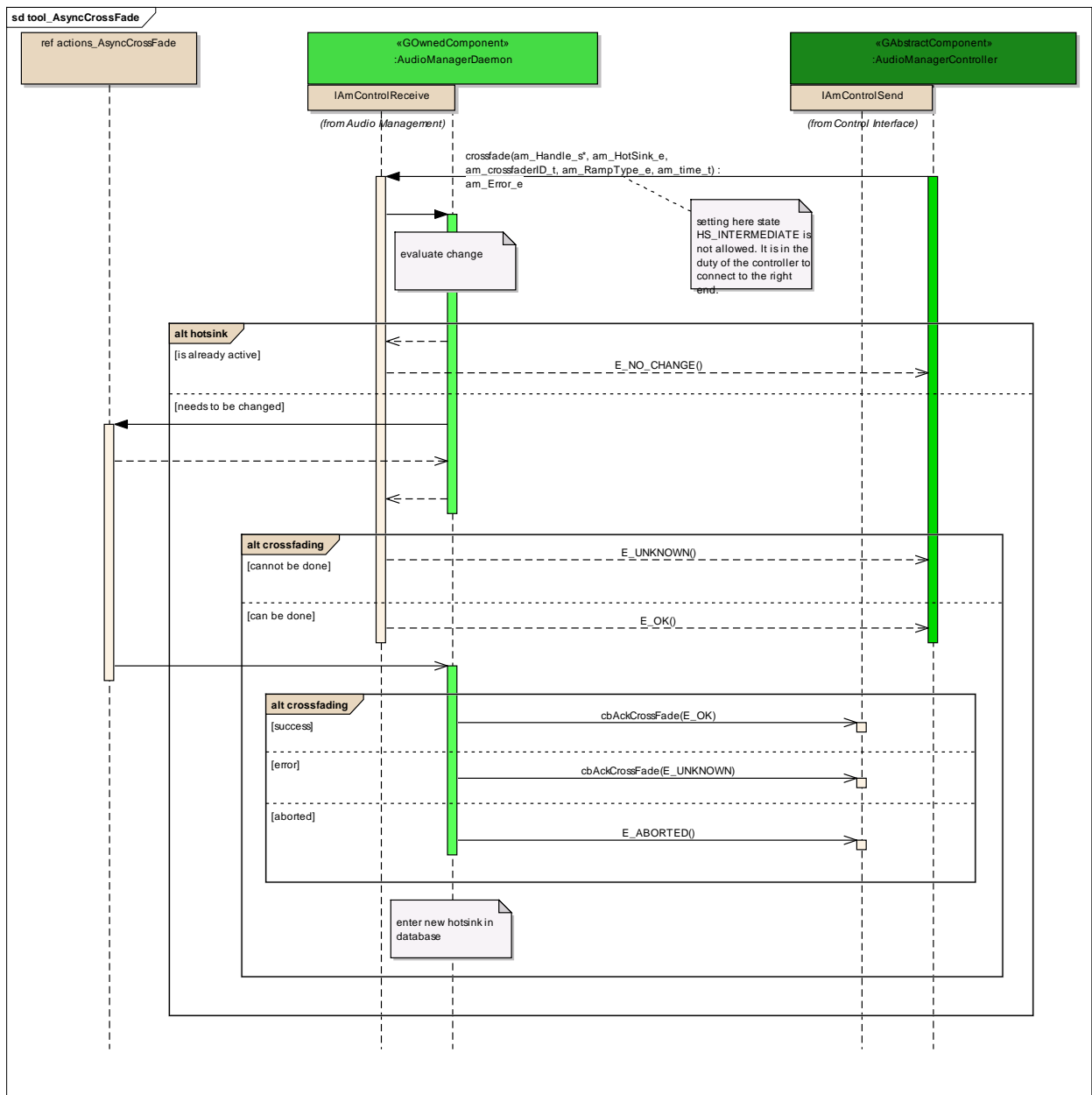
Figure: 14

actions_AsyncCrossFade

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncCrossFade *Keywords:*
Detail:	Created on 12.10.2011. Last modified on 12.10.2011.
GUID:	{FBF0561A-3DA6-451e-95A4-496CDD2CC299}

Connector	Source	Target	Notes
Sequence Source -> Destination	Public <anonymous>	Public actions_AsyncCrossFade	
Sequence Source -> Destination	Public actions_AsyncCrossFade	Public <anonymous>	
Sequence Source -> Destination	Public actions_AsyncCrossFade	Public <anonymous>	

crossfading

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncCrossFade *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{62EEC252-B1D7-4af3-8FAD-961FC1A162B8}

crossfading

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncCrossFade *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{3B973F2A-A377-495f-B131-138860EC255A}

hotsink

Type:	**InteractionFragment**

Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncCrossFade *Keywords:*
Detail:	*Created on 07.09.2011. Last modified on 12.10.2011.*
GUID:	{9C6425E5-D5DE-4550-80C3-66854DC3469E}

r_AsyncCrossFade

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_AsyncCrossFade *Keywords:*
Detail:	*Created on 07.09.2011. Last modified on 07.09.2011.*
GUID:	{C4EC5D13-1050-42b4-8409-B1CDEAD073F9}

tool_SetCommandReady

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011*
GUID:	{79C5CCAE-413B-4b59-839A-EC2521B811CF}

tool_SetCommandReady - *(Interaction diagram)*

Created By:	christian *on* 12.10.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{58360405-E6EA-479f-857B-3D9D4B9B2E01}
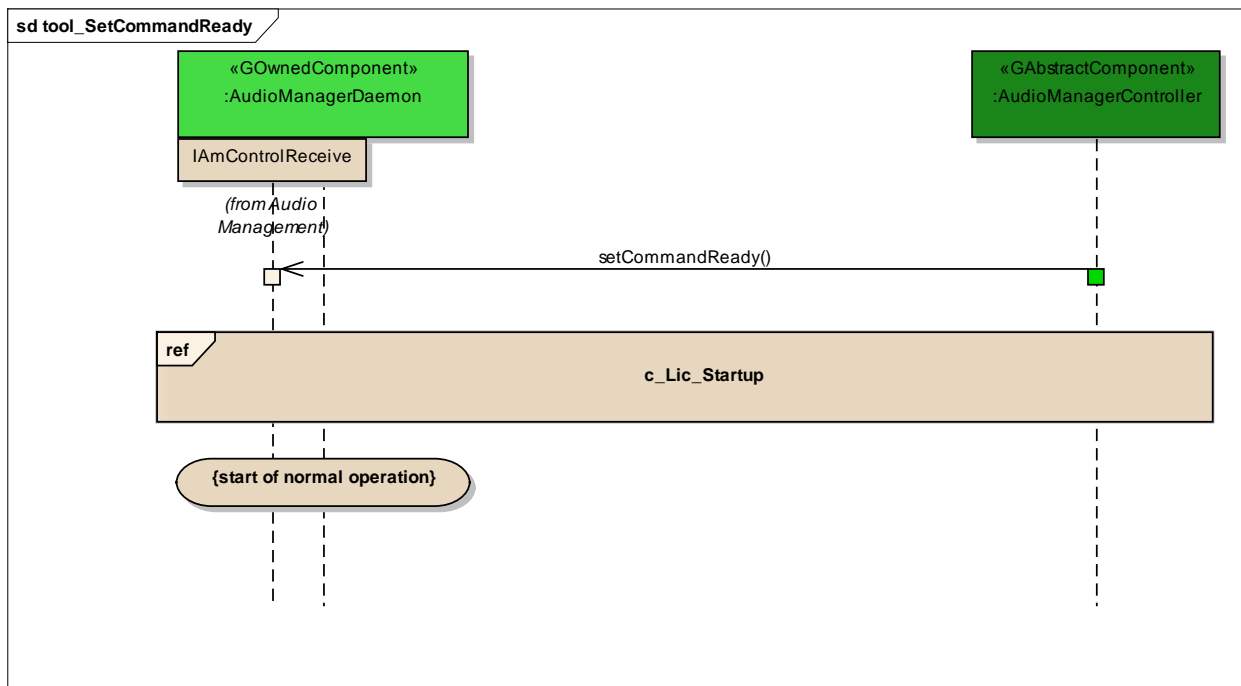
Figure: 15

c_Lic_Startup

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_SetCommandReady *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{81F765B0-9153-4168-9A2C-9C397B77E7F0}

tool_SetRoutingReady

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011*
GUID:	{13F97391-A629-4638-99F5-A0ACDB1124E7}

tool_SetRoutingReady - *(Interaction diagram)*
Created By: christian *on* 12.10.2011

Last Modified: 12.10.2011
Version: 1.0. *Locked:* False
GUID: {99BED923-360E-4df4-AD6F-74DDBFB26E48}
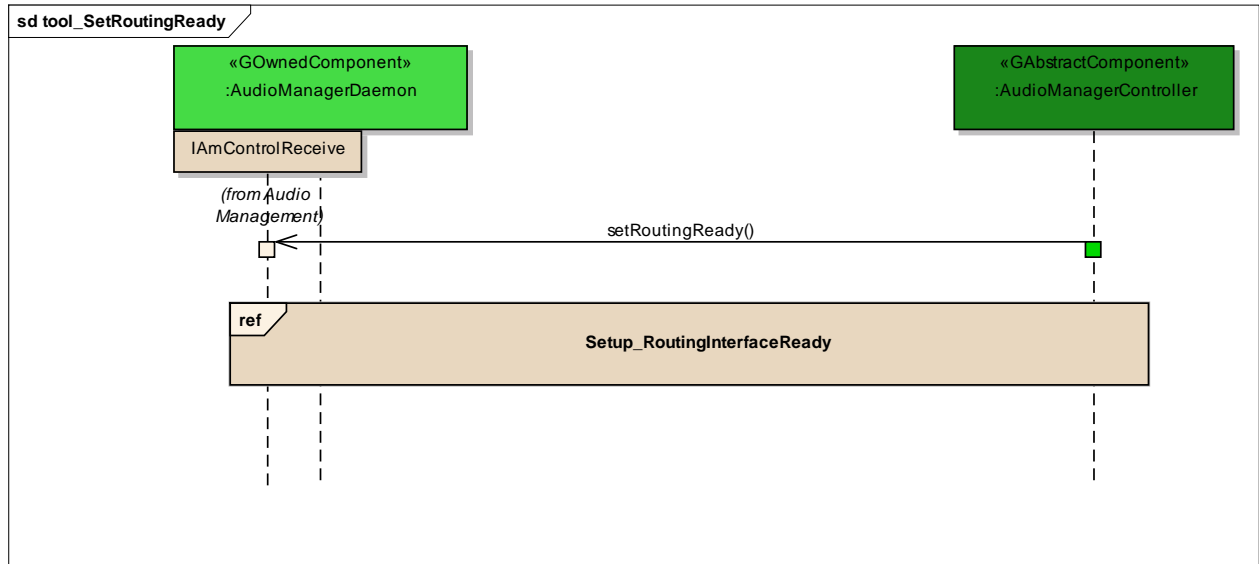
Figure: 16

Setup_RoutingInterfaceReady

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_SetRoutingReady *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{F5EE53F3-F5DF-44ed-9DA0-5060AD7C21A2}

tool_SetDomainState

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 11.10.2011. Last modified on 12.10.2011*
GUID:	{C51657D3-73EA-4f53-9829-AA4D12E8B18B}

tool_SetDomainState - *(Interaction diagram)*
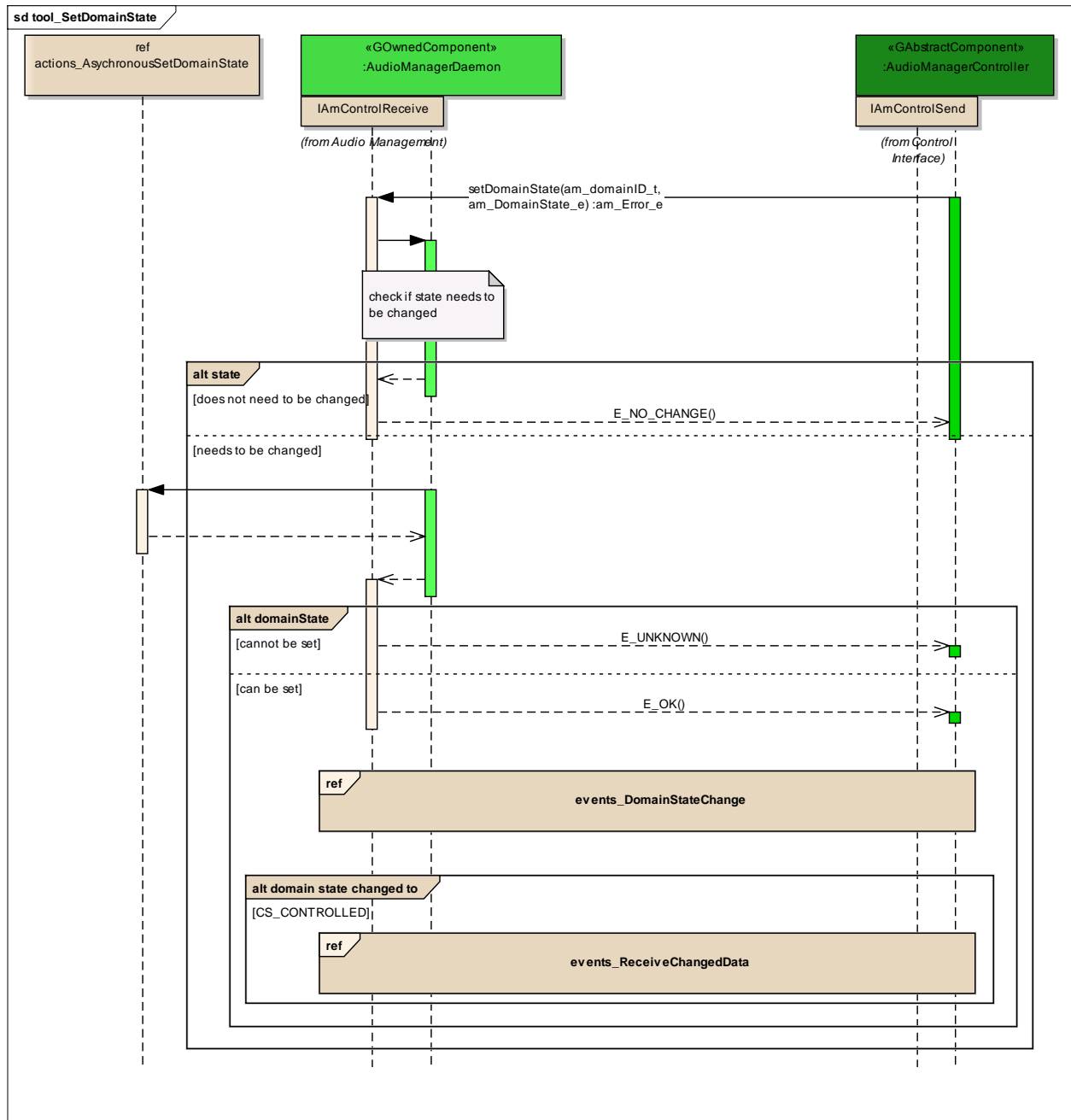
Figure: 17

actions_AsychronousSetDomainState

Type:	**InteractionOccurrence**	
Status:	Proposed. Version 1.0. Phase 1.0.	
Package:	tool_SetDomainState	*Keywords:*
Detail:	Created on 11.10.2011. Last modified on 11.10.2011.	
GUID:	{3A60057F-7D23-4284-9E93-165BBE3B82C6}	

Connections

Connector	Source	Target	Notes
Sequence Source -> Destination	Public <anonymous>	Public actions_Asychrono usSetDomainState	
Sequence Source -> Destination	Public actions_Asychrono usSetDomainState	Public <anonymous>	

domain state changed to

Type:	**InteractionFragment**	
Status:	Proposed. Version 1.0. Phase 1.0.	
Package:	tool_SetDomainState	*Keywords:*
Detail:	Created on 12.10.2011. Last modified on 12.10.2011.	
GUID:	{A74F546F-F328-4f29-AC2A-90A0B926147E}	

domainState

Type:	**InteractionFragment**	
Status:	Proposed. Version 1.0. Phase 1.0.	
Package:	tool_SetDomainState	*Keywords:*
Detail:	Created on 11.10.2011. Last modified on 11.10.2011.	
GUID:	{3A283AFD-64C4-4813-A213-11645863518F}	

events_DomainStateChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_SetDomainState *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{B8B2E07F-D7D3-44b6-BDEF-F98645068689}

events_ReceiveChangedData

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_SetDomainState *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{9CCEE126-22E4-46e6-8E91-65F31BE156DE}

state

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	tool_SetDomainState *Keywords:*
Detail:	*Created on 11.10.2011. Last modified on 11.10.2011.*
GUID:	{97873870-46BF-4e72-A29A-0B80696C5E08}

tool_AsyncSetMultipleVolumes

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Toolbox
Detail:	*Created on 29.11.2012. Last modified on 29.11.2012*
GUID:	{68383A42-EF61-431a-9293-1CB97DA9BE20}

tool_AsyncSetMultipleVolumes - *(Interaction diagram)*

Created By:	q232968 *on* 29.11.2012
Last Modified:	29.11.2012
Version:	1.0. *Locked:* False
GUID:	{C75427B7-E8C3-481a-AFB5-F96F8C289F3E}
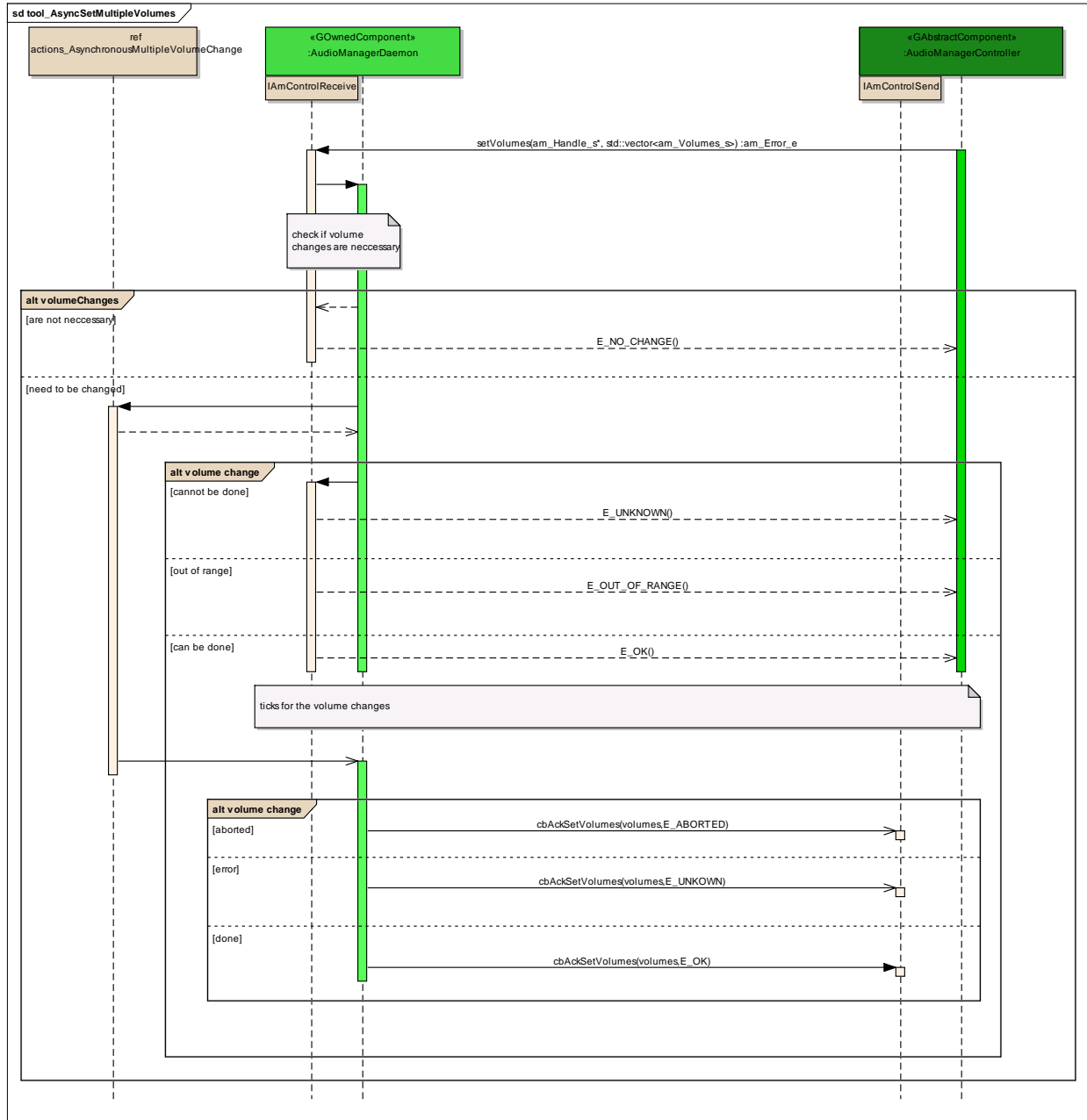
Figure: 18

actions_AsynchronousMultipleVolumeChange

Type:	**InteractionOccurrence**	
Status:	Proposed. Version 1.0. Phase 1.0.	
Package:	tool_AsyncSetMultipleVolumes	*Keywords:*
Detail:	*Created on 29.11.2012. Last modified on 29.11.2012.*	
GUID:	{073E11BA-0CCD-451f-A4D3-3B40E85DB99D}	

Connections

Connector	Source	Target	Notes
Sequence Source -> Destination	Public <anonymous>	Public actions_Asynchron ousMultipleVolume Change	
Sequence Source -> Destination	Public actions_Asynchron ousMultipleVolume Change	Public <anonymous>	
Sequence Source -> Destination	Public actions_Asynchron ousMultipleVolume Change	Public <anonymous>	

volume change

Type:	**InteractionFragment**	
Status:	Proposed. Version 1.0. Phase 1.0.	
Package:	tool_AsyncSetMultipleVolumes	*Keywords:*
Detail:	*Created on 29.11.2012. Last modified on 29.11.2012.*	
GUID:	{A0837C6B-F535-4d52-BEAC-FB63C3F25373}	

volume change

Type:	**InteractionFragment**	
Status:	Proposed. Version 1.0. Phase 1.0.	
Package:	tool_AsyncSetMultipleVolumes	*Keywords:*

volumeChanges

Type: **InteractionFragment**
Status: Proposed. Version 1.0. Phase 1.0.
Package: tool_AsyncSetMultipleVolumes   *Keywords:*
Detail: *Created on 29.11.2012. Last modified on 29.11.2012.*
GUID: {3D1BD540-F1DE-43e8-8E36-D46E25267817}

Service
Type: **Package**
Status: Reworked. Version 1.0. Phase 1.0.
Package: Control Interface
Detail: *Created on 22.09.2011. Last modified on 10.10.2011*
GUID: {1BAE56DF-5A2C-40cc-9868-1B50C23827AB}

Node State Manager

Type: **GNamedPlaceholder**
Status: Proposed. Version 1.0. Phase 1.0.
Package: Service   *Keywords:*
Detail: *Created on 22.02.2012. Last modified on 22.02.2012.*
GUID: {5DCAC729-4461-4ba0-9F18-1C83184F4DB4}

This is a placeholder for the GENIVI Node State Manager that is about to come.

Custom Properties

- isIndirectlyInstantiated = True

service_GetRoute

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Service
Detail:	*Created on 22.09.2011. Last modified on 10.10.2011*
GUID:	{5215F021-E9AA-437f-989C-746C3754B857}

Behind this sequence, the autorouting service of the audiomanger deamon is shown.

service_GetRoute - *(Interaction diagram)*

Created By:	christian *on* 22.09.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{F5B1809D-ADE3-413f-8FC1-351E8913DF31}
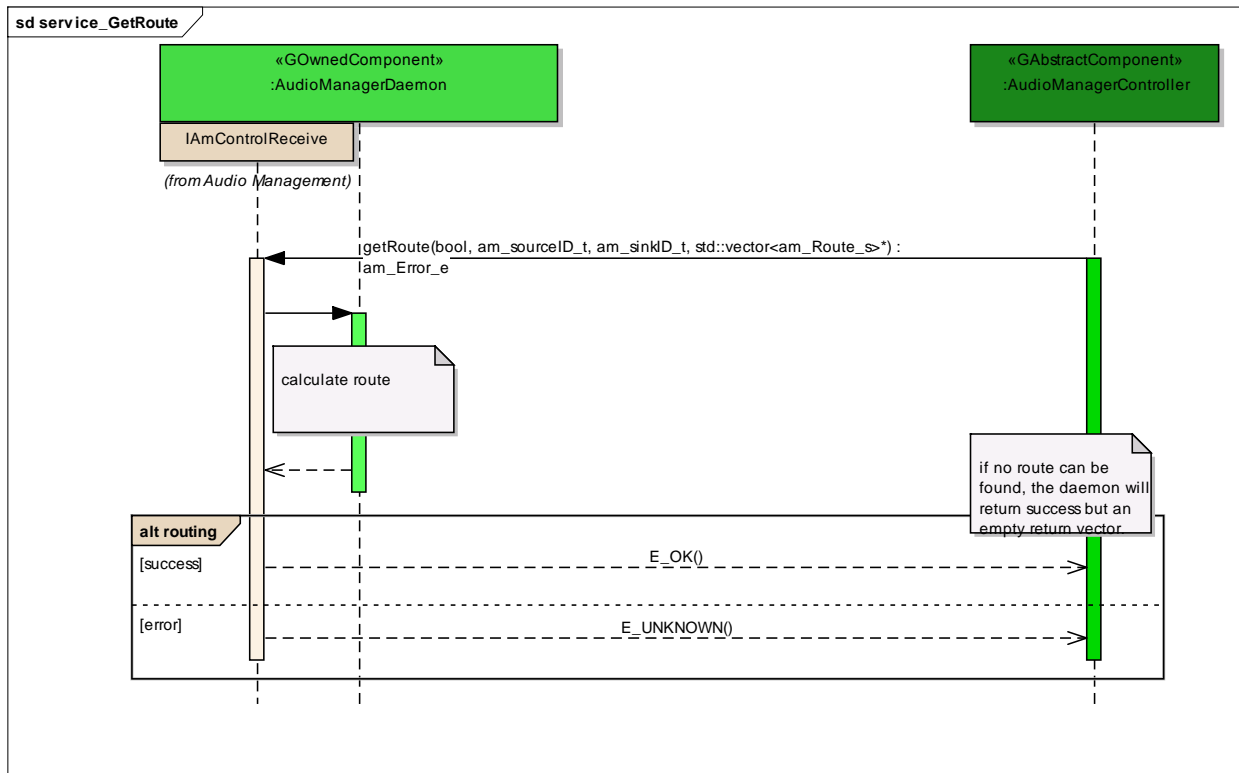
Figure: 19

routing

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	service_GetRoute *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{D240B1AF-A2B8-405f-B4EC-FE450233D913}

service_CancelRundown

Type:	**Package**
Status:	. Version 1.0. Phase 1.0.
Package:	Service
Detail:	Created on 22.02.2012. Last modified on 22.02.2012
GUID:	{9FD618B2-CAEB-4d12-9FD6-837861E6BFAE}

service_CancelRundown - *(Interaction diagram)*

Created By:	christian *on* 12.10.2011
Last Modified:	23.02.2012
Version:	1.0. *Locked:* False
GUID:	{92B355F5-9264-4a72-A2A6-F18AF05A3517}
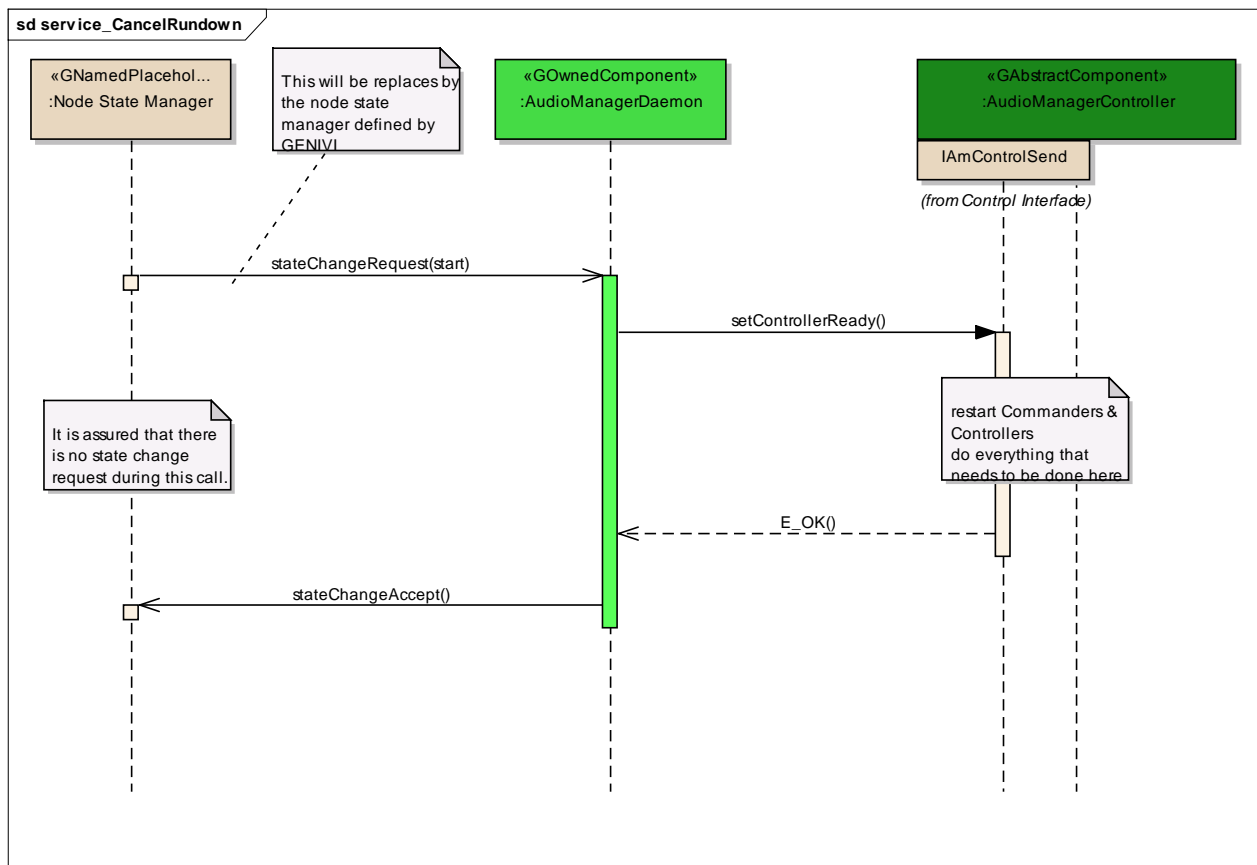
Figure: 20

<anonymous>

Type:	**GNamedPlaceholder**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	service_CancelRundown *Keywords:*
Detail:	*Created on 22.02.2012. Last modified on 22.02.2012.*
GUID:	{2BF5D262-876F-4aca-AB33-9A989A1D18C2}

Custom Properties

- isIndirectlyInstantiated = True

Connections

Connector	Source	Target	Notes
NoteLink	Public <anonymous>	Public <anonymous>	
Sequence stateChangeRequest Source -> Destination	Public <anonymous>	Public <anonymous>	
Sequence stateChangeAccept Source -> Destination	Public <anonymous>	Public <anonymous>	

initialization

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	service_CancelRundown *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{1B64A8BC-2171-472f-81A6-7AA77CD1AA67}

service_Startup

Type:	**Package**

Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Service
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011*
GUID:	{ADBA613D-4C35-450c-9AD5-43DE8FBA42C4}

service_Startup - *(Interaction diagram)*

Created By:	christian *on* 12.10.2011
Last Modified:	22.02.2012
Version:	1.0. *Locked:* False
GUID:	{0C0A015B-4788-4e3c-9946-E6CD28EF8D22}
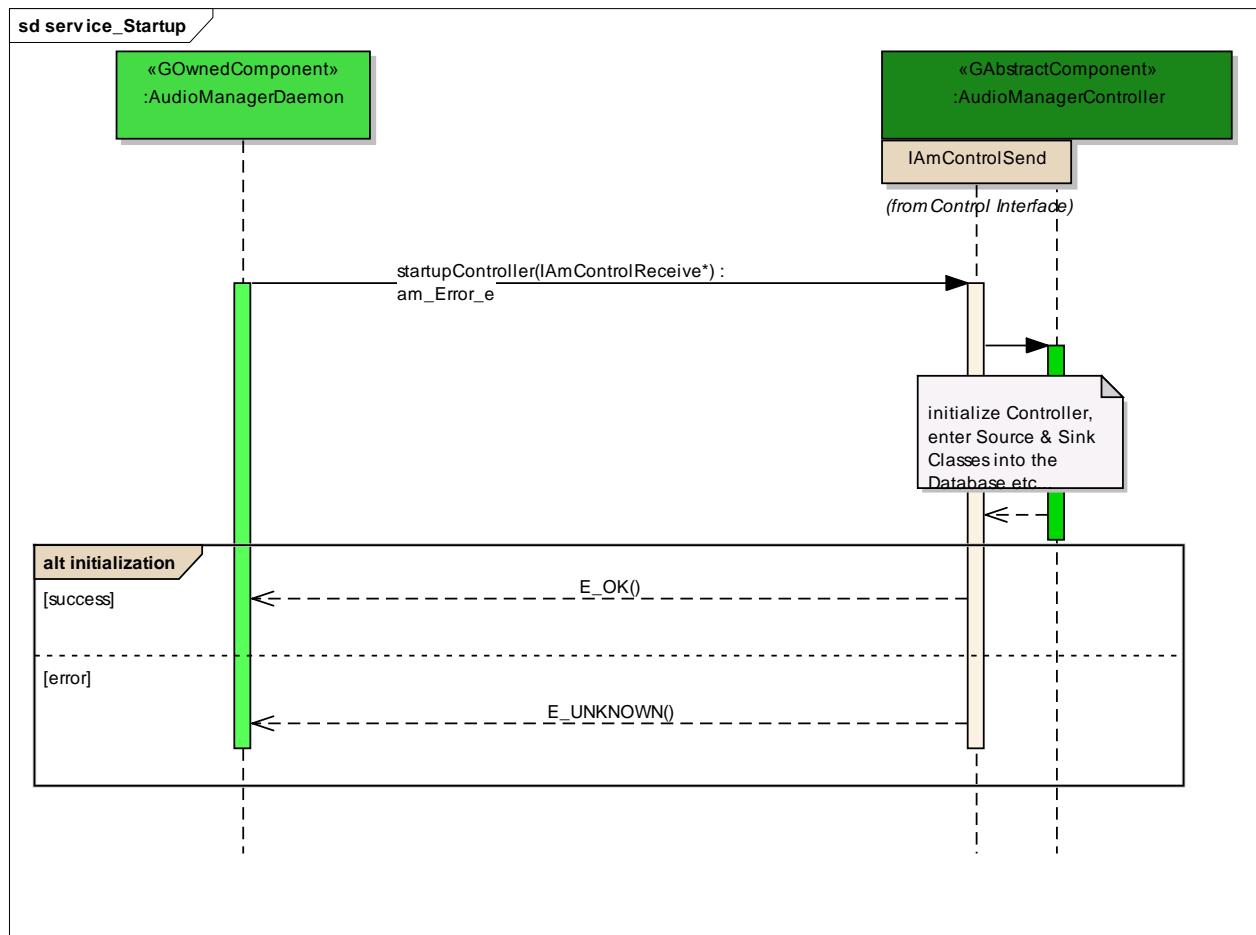
Figure: 21

initialization

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	service_Startup *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*

GUID: {0F679475-F010-4b2c-A30A-9198A440E2FA}

service_Rundown

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Service
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011*
GUID:	{5DD7E440-08E8-42c6-9BDC-E5EED1B7650F}

service_Rundown - *(Interaction diagram)*

Created By:	christian *on* 12.10.2011
Last Modified:	22.02.2012
Version:	1.0. *Locked:* False
GUID:	{C1592DFF-A368-4b9e-AEFC-8FEFD788262F}
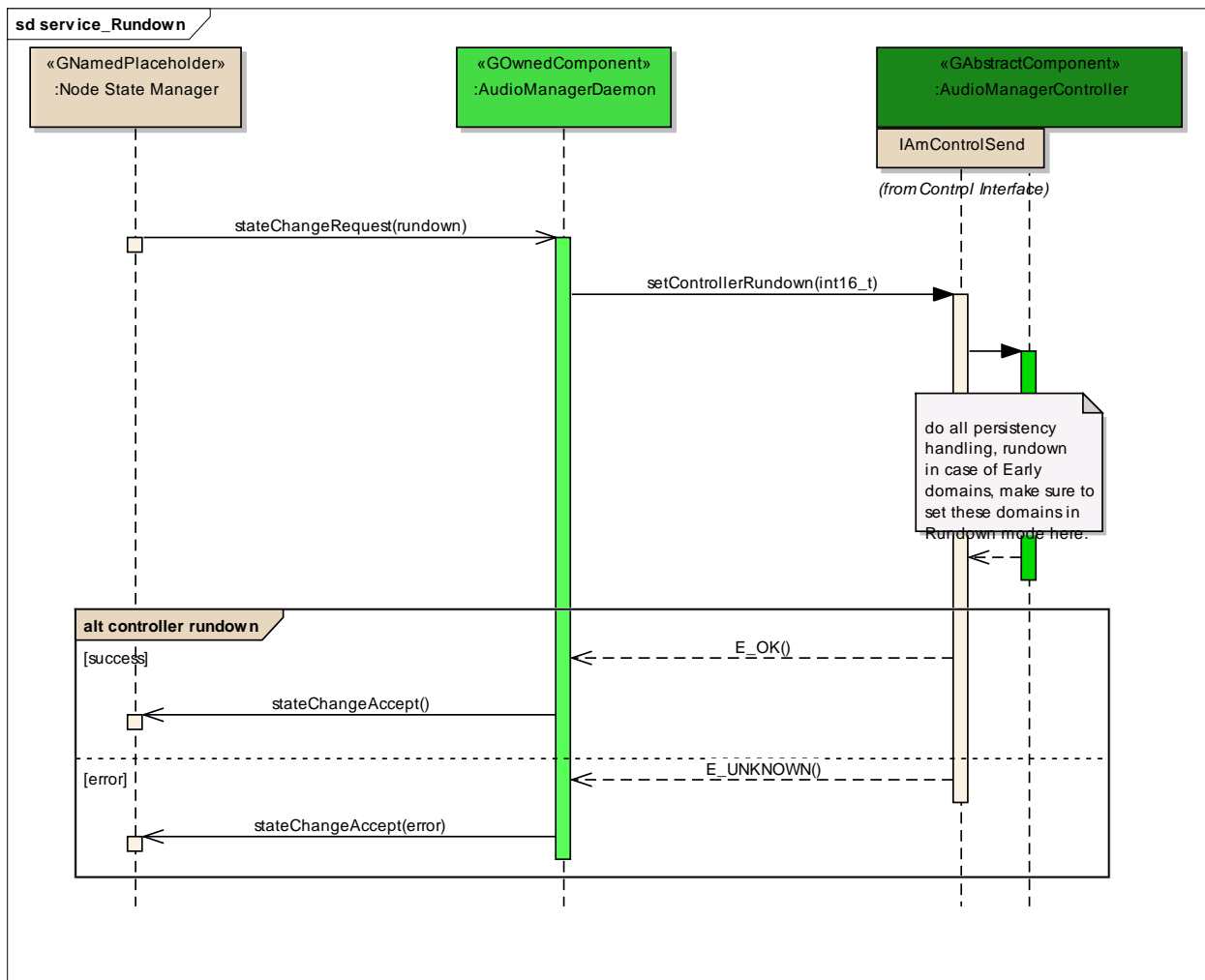
Figure: 22

<anonymous>

Type:	**GNamedPlaceholder**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	service_Rundown *Keywords:*
Detail:	Created on 22.02.2012. Last modified on 22.02.2012.
GUID:	{7B307B0C-F56D-441f-BC45-ADBF6FF3BD9B}

Custom Properties
- isIndirectlyInstantiated = True

Connector	Source	Target	Notes
Sequence stateChangeRequest Source -> Destination	Public <anonymous>	Public <anonymous>	
Sequence stateChangeAccept Source -> Destination	Public <anonymous>	Public <anonymous>	
Sequence stateChangeAccept Source -> Destination	Public <anonymous>	Public <anonymous>	

controller rundown

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	service_Rundown *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{83880219-5957-4065-8508-E1642F207A2B}

9.2 Database Access

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Control Interface
Detail:	*Created on 13.09.2011. Last modified on 13.10.2011*
GUID:	{1099289F-7DE6-4034-9DC3-3134E58A85A5}

All sequences below hold information for database access

database_ChangeMainSinkNotification

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access

database_ChangeMainSinkNotification - *(Interaction diagram)*

Created By:	q232968 *on* 29.01.2013
Last Modified:	29.01.2013
Version:	1.0. *Locked:* False
GUID:	{E85D4596-0EBB-4738-9A1F-798E63B8DCDC}
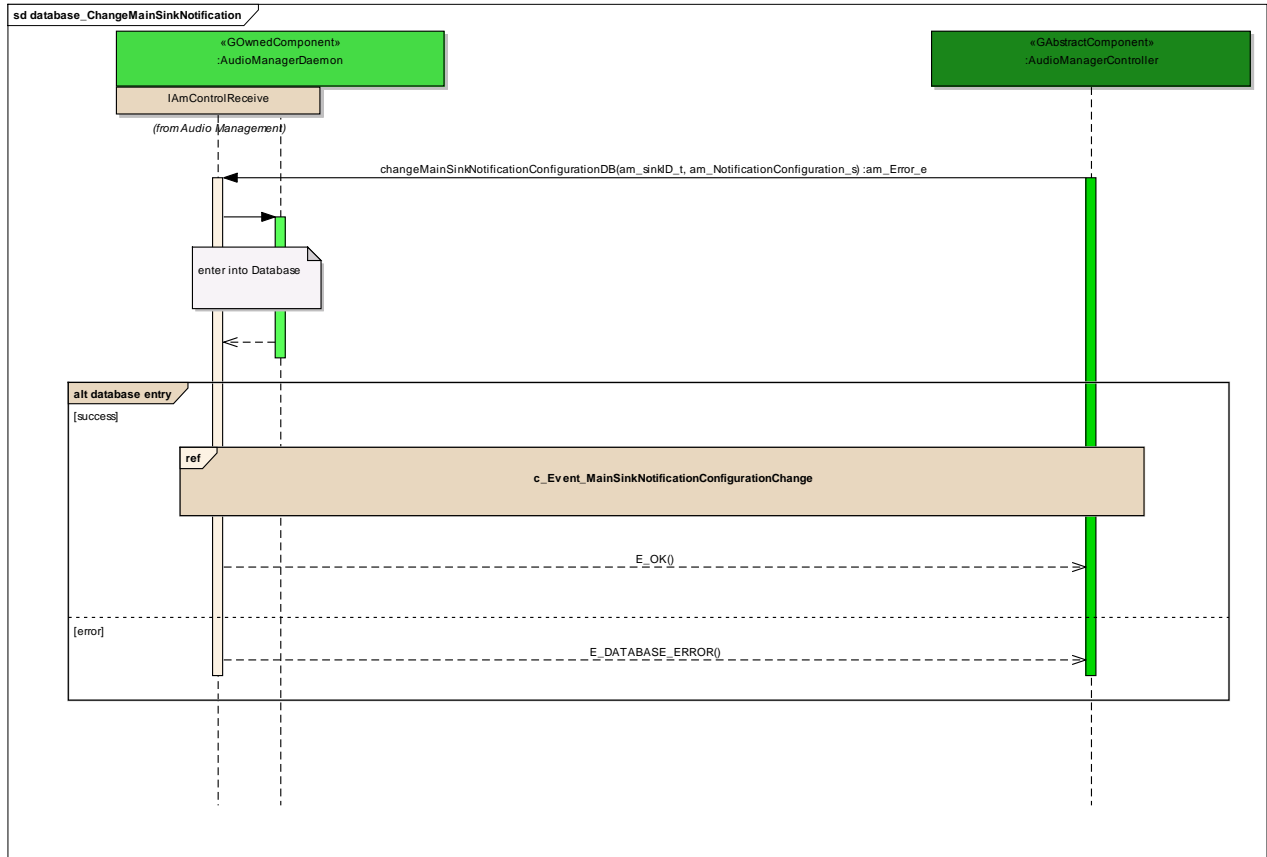
Figure: 23

c_Event_MainSinkNotificationConfigurationChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMainSinkNotification *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{AF6AE439-3E07-4524-96A8-E2D190E5E7A8}

database entry

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMainSinkNotification *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{179601A2-E060-4572-A8D0-6F3BB70A096B}

database_ChangeMainSourceNotification

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013*
GUID:	{FC703BB5-0C5F-4931-B132-8A161AEDDC57}

database_ChangeMainSourceNotification - *(Interaction diagram)*

Created By:	q232968 *on* 29.01.2013
Last Modified:	29.01.2013
Version:	1.0. *Locked:* False
GUID:	{50538E92-144C-4a12-ACAC-C674C0006931}
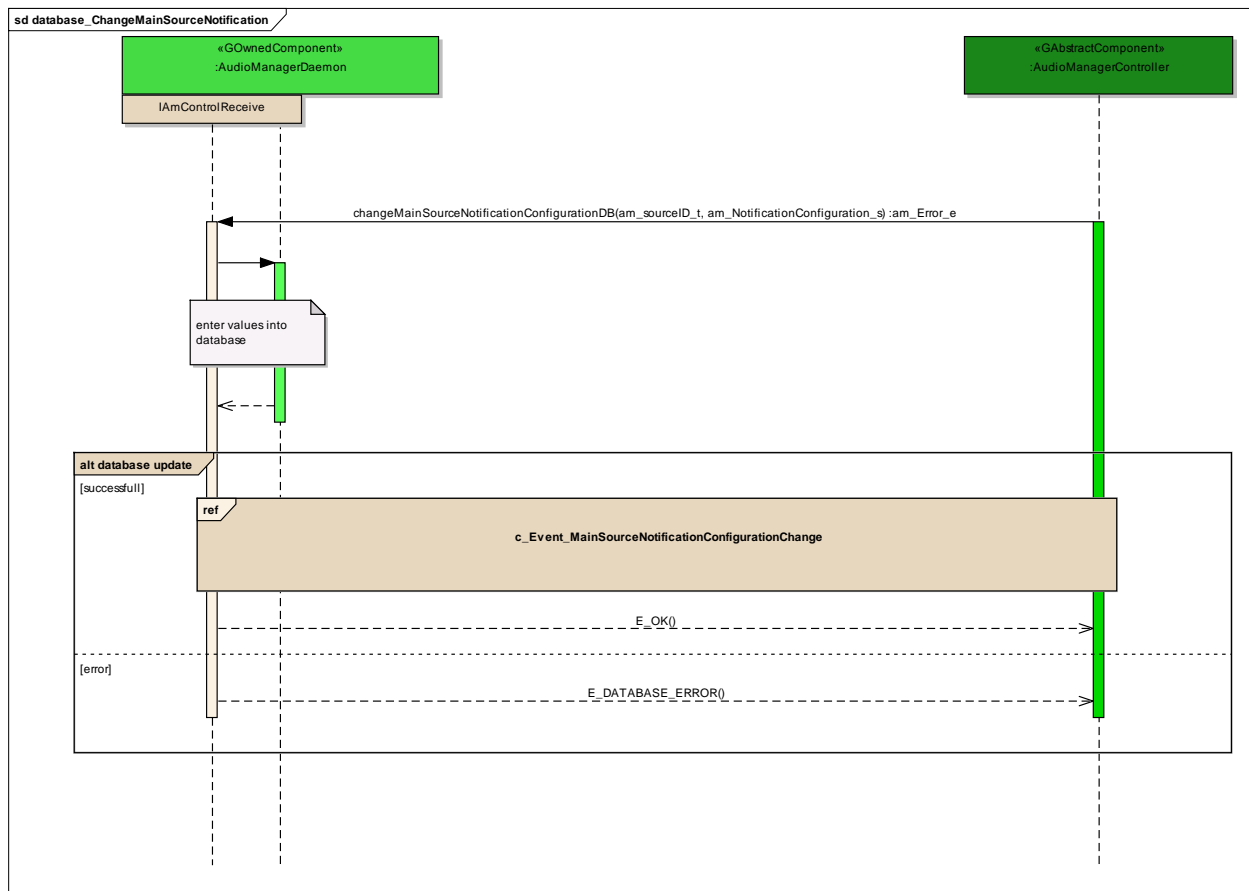
Figure: 24

c_Event_MainSourceNotificationConfigurationChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMainSourceNotification *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{218CD7FE-5273-4b77-811E-8179AF0E658E}

database update

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMainSourceNotification *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{94982892-571B-48e4-A322-2362453C33C2}

database_GetListCrossfadersOfDomain

Type: **Package**
Status: Proposed. Version 1.0. Phase 1.0.
Package: Database Access
Detail: *Created on 13.10.2011. Last modified on 13.10.2011*
GUID: {3F482C6F-F9F5-4cd2-A403-EBCC15D24659}

database_GetListCrossfadersOfDomain - *(Interaction diagram)*

Created By: christian *on* 13.10.2011
Last Modified: 13.10.2011
Version: 1.0. *Locked:* False
GUID: {340C48FF-C0C3-4d17-ADAF-5B8A5DC1A4C0}
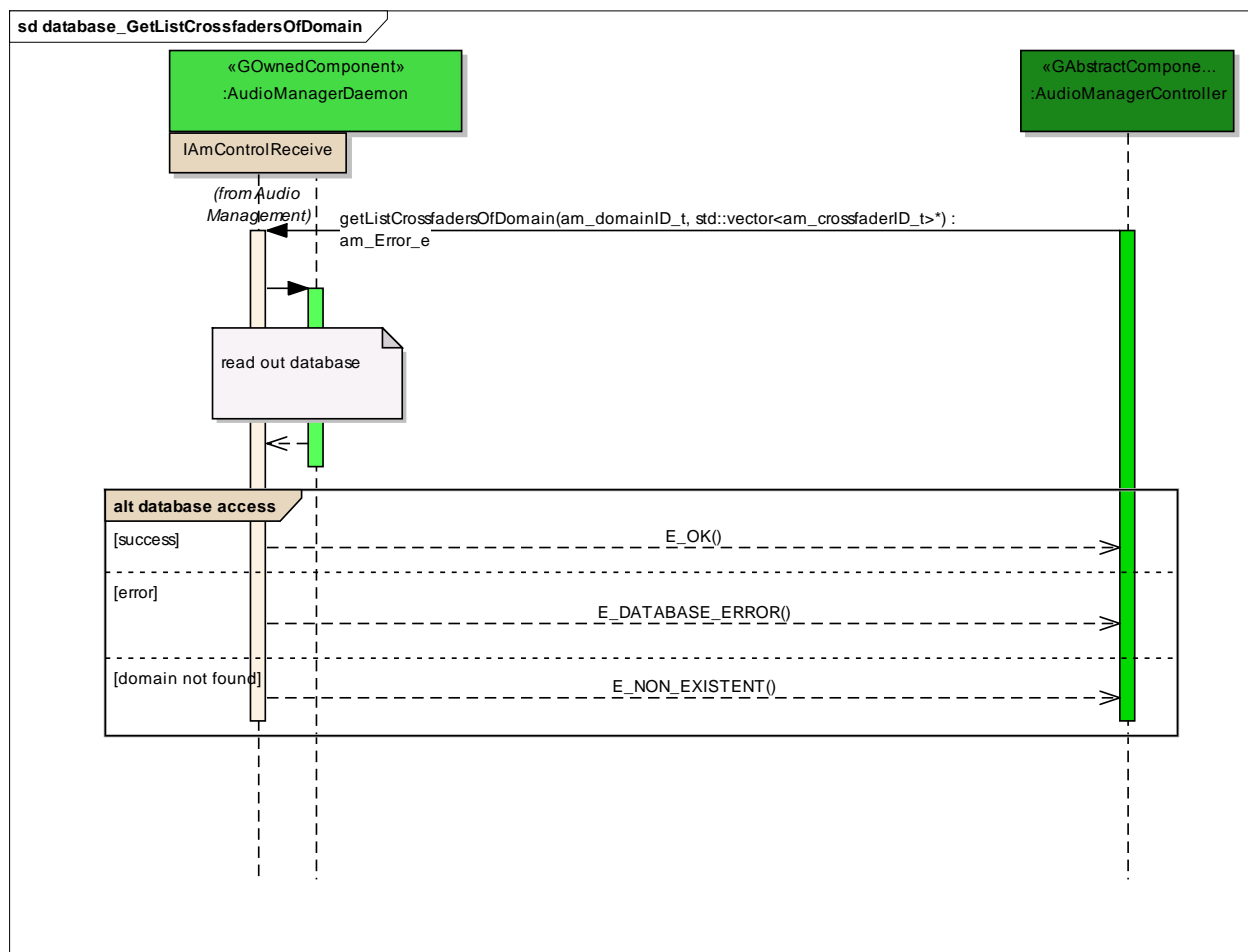
Figure: 25

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListCrossfadersOfDomain *Keywords:*
Detail:	*Created on 13.10.2011. Last modified on 13.10.2011.*
GUID:	{D84EC7B5-61F1-4c2f-9BA6-6F8E66E10C31}

database_GetSourceSinkOfGateway

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.

Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{C3225F03-AC0E-4736-B0F0-EC87E7829CBB}

database_GetSourceSinkOfGateway - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{5A3480C6-7CD0-424f-B94A-E4F50B9360B2}
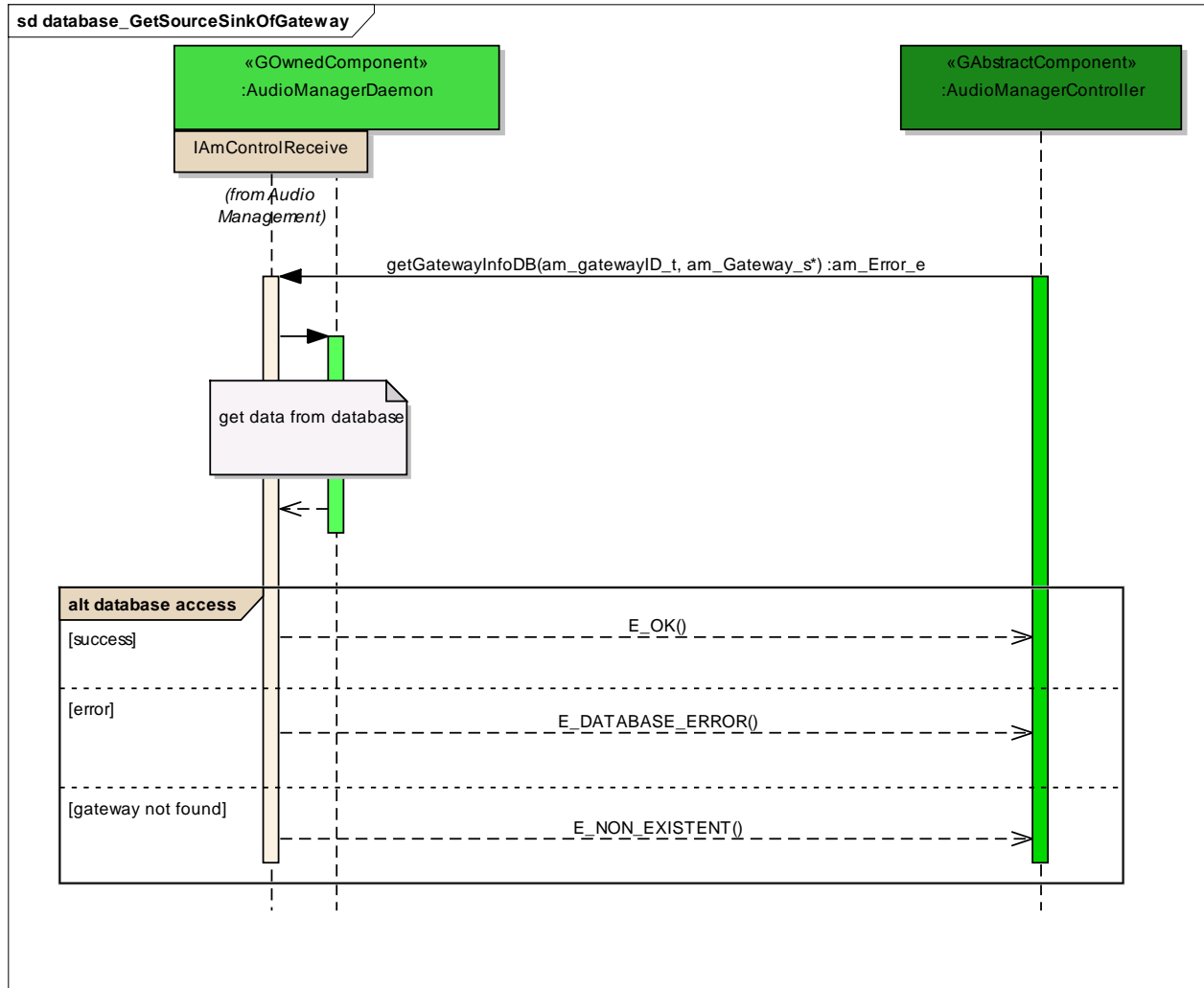
Figure: 26

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.

Package:	database_GetSourceSinkOfGateway	*Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*	
GUID:	{9842CB15-5CB7-4028-BD32-EE4D79AD812D}	

database_GetSourceSinksOfCrossfader

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011*
GUID:	{2419DE40-B4A6-4277-9845-9EB25C7D5789}

database_GetSourceSinksOfCrossfader - *(Interaction diagram)*

Created By:	christian *on* 12.10.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{E0D8A066-DFAB-438f-8EE0-D95CF4854AD2}
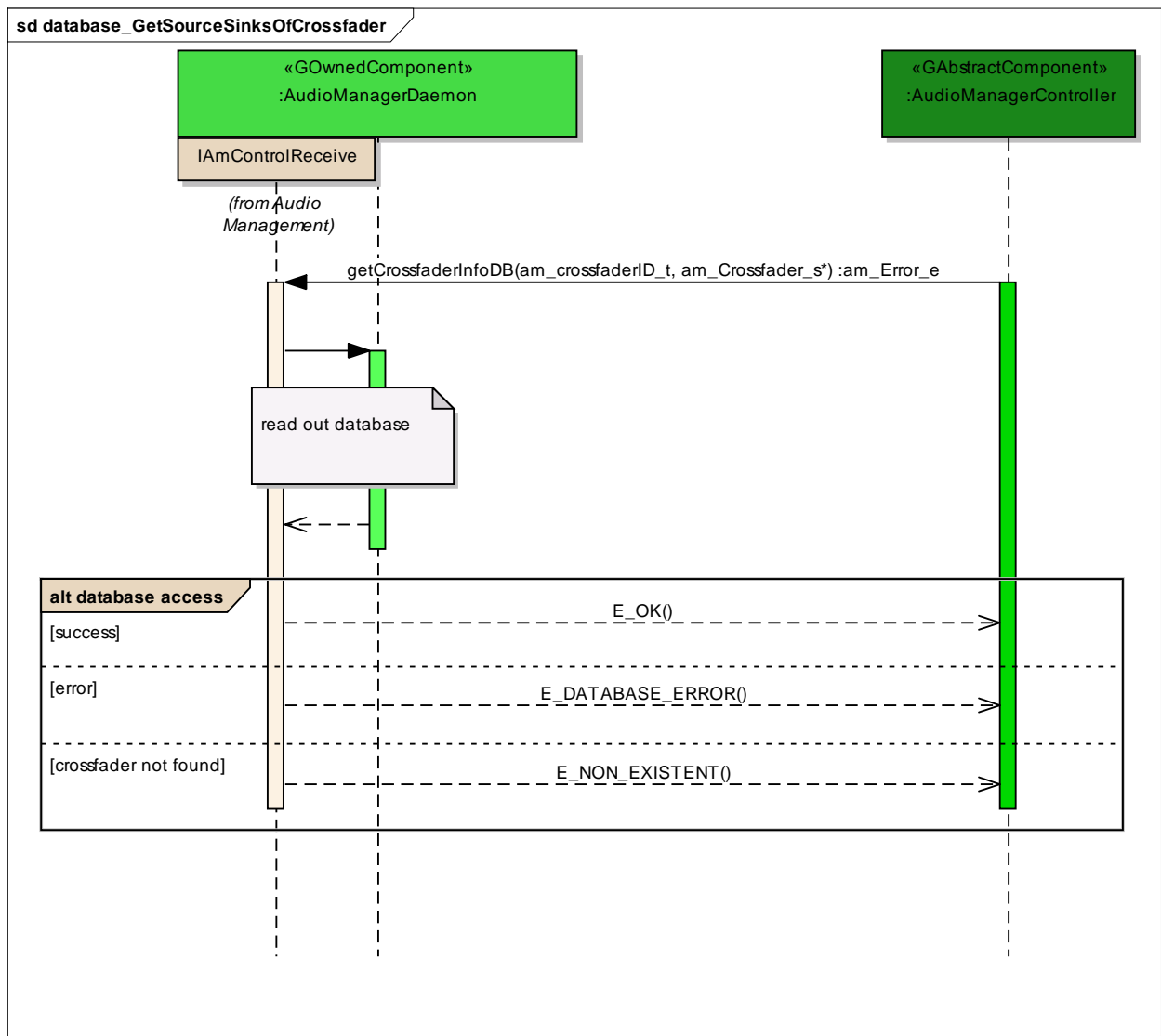
Figure: 27

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetSourceSinksOfCrossfader *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{BCB5792D-8BB3-43f6-8755-DB91463D1C33}

database_GetListGatewaysOfDomain

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{5C44ACA5-D7E8-4bfc-BDE5-6037F45B8856}

database_GetListGatewaysOfDomain - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{BA0D4FC4-DB6F-43c8-A8E0-33FFC1A22E32}
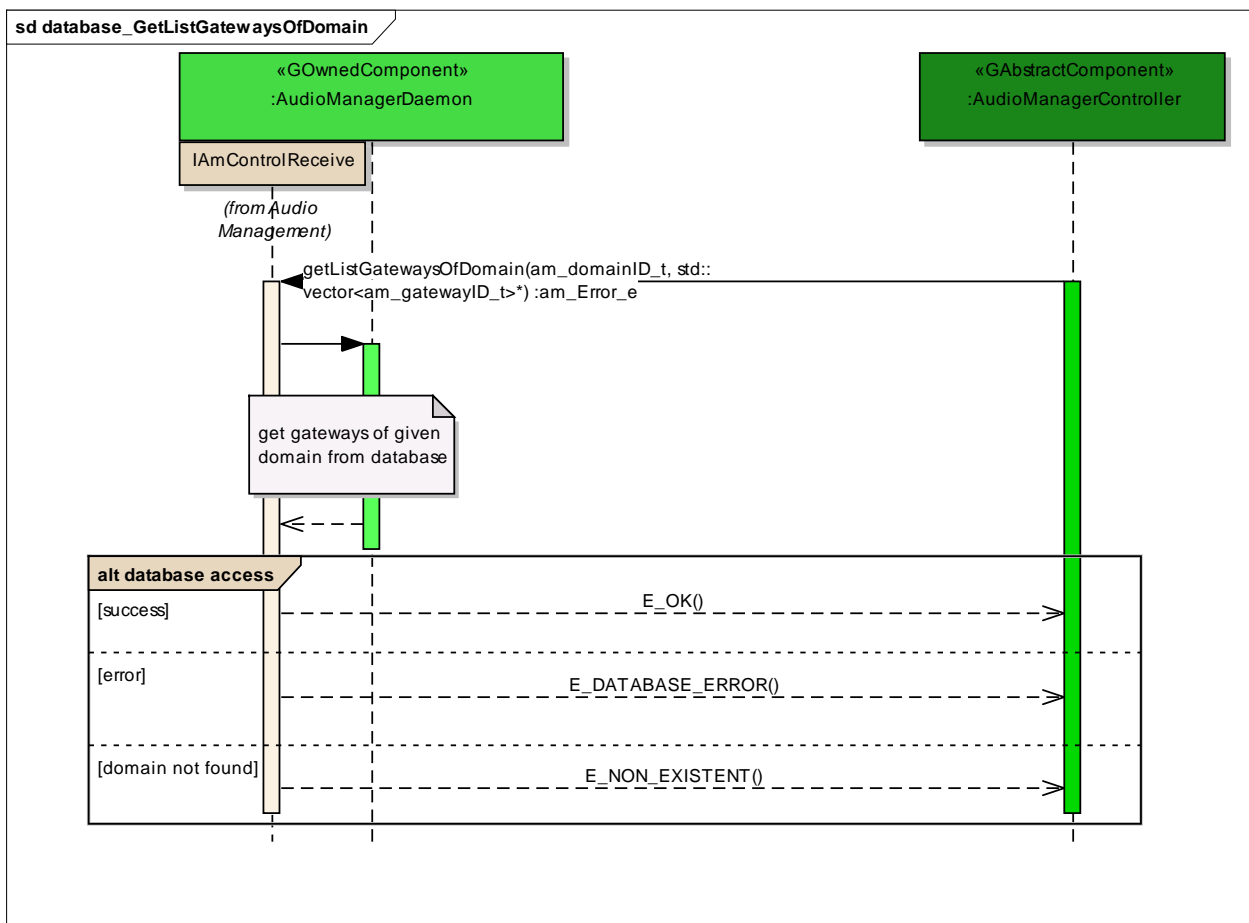
Figure: 28

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.

Package: database_GetListGatewaysOfDomain   *Keywords:*
Detail: *Created on 10.10.2011. Last modified on 10.10.2011.*
GUID: {EC288E28-CF1F-4631-BFE0-2930DA5A70E0}

database_GetListGateways

Type: **Package**
Status: Reworked. Version 1.0. Phase 1.0.
Package: Database Access
Detail: *Created on 23.09.2011. Last modified on 07.10.2011*
GUID: {19C93F63-4016-4509-B8B5-077A1CE35263}

database_GetListGateways - *(Interaction diagram)*

Created By: christian *on* 23.09.2011
Last Modified: 07.10.2011
Version: 1.0. *Locked:* False
GUID: {0E719069-9DDF-4324-B596-6A5CA1FCE779}
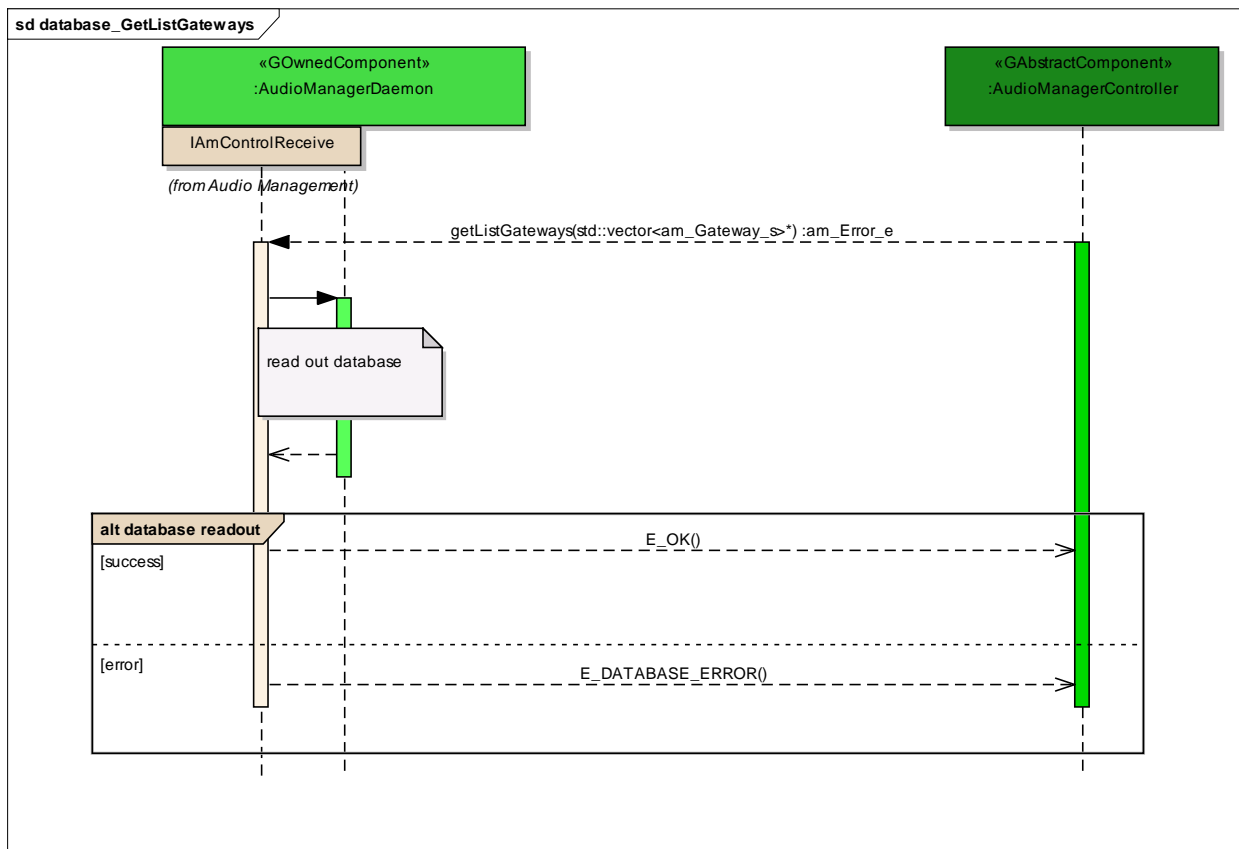
Figure: 29

database readout

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListGateways *Keywords:*
Detail:	*Created on 07.10.2011. Last modified on 07.10.2011.*
GUID:	{94C3D121-F8B8-46c0-B26E-49D1E2A3EBC2}

database_GetListCrossfaders

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011*
GUID:	{CA1E51D5-3942-44ac-88F7-72495E1F69FA}

database_GetListCrossfaders - *(Interaction diagram)*

Created By: christian *on* 12.10.2011
Last Modified: 12.10.2011
Version: 1.0. *Locked:* False
GUID: {4BC71AEA-687D-470c-8B09-7C886DE6E8C3}
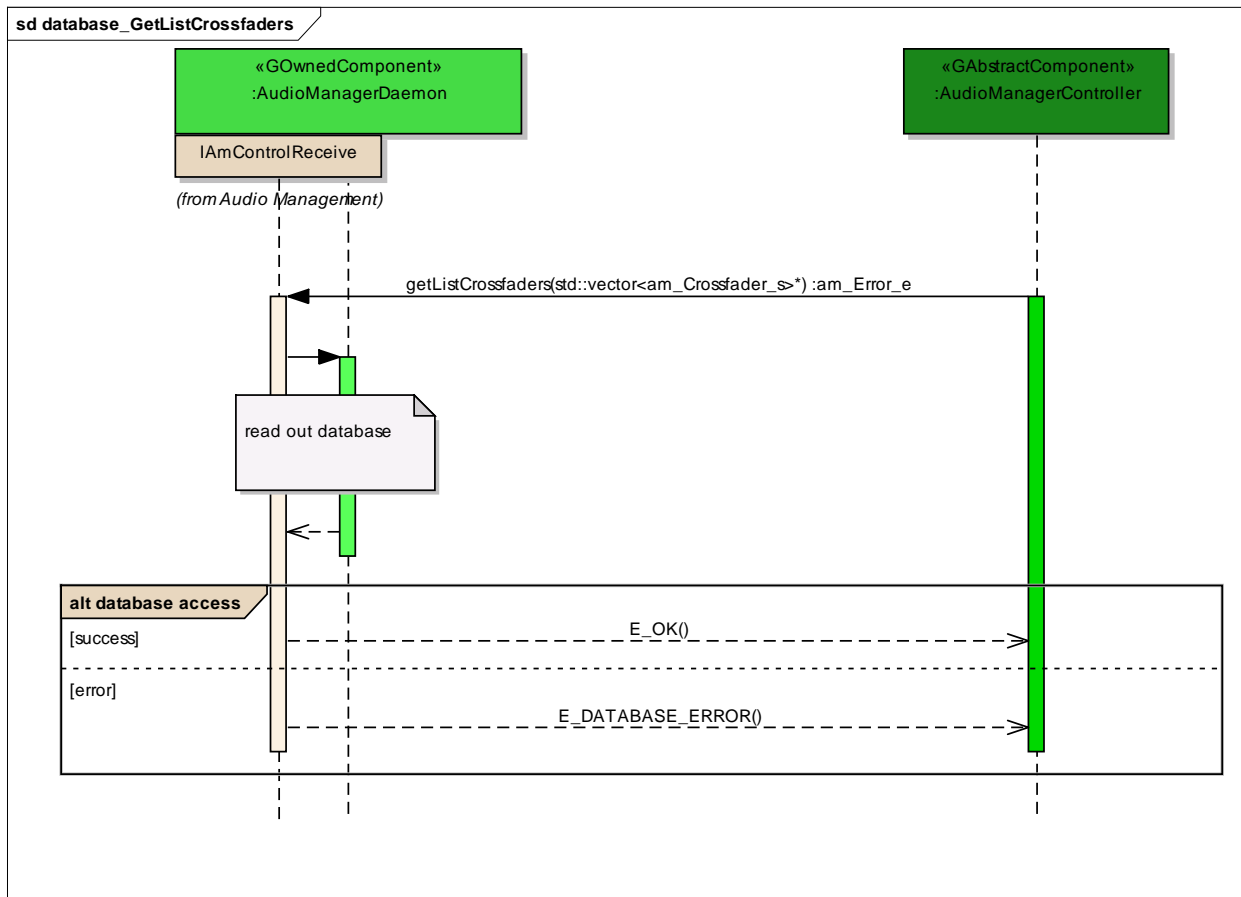
Figure: 30

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListCrossfaders *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{A8CA58C2-3A27-4f4d-AD2D-F7AE7FEF970B}

database_GetListHandles

Type: **Package**
Status: Reworked. Version 1.0. Phase 1.0.
Package: Database Access
Detail: *Created on 23.09.2011. Last modified on 07.10.2011*
GUID: {2A45DE39-0343-4502-B4C5-DFE7EFDBC546}

database_GetListHandles - *(Interaction diagram)*

Created By: christian *on* 23.09.2011
Last Modified: 07.10.2011
Version: 1.0. *Locked:* False
GUID: {55A24D50-1432-43b2-9B1A-43DC8AC89D21}
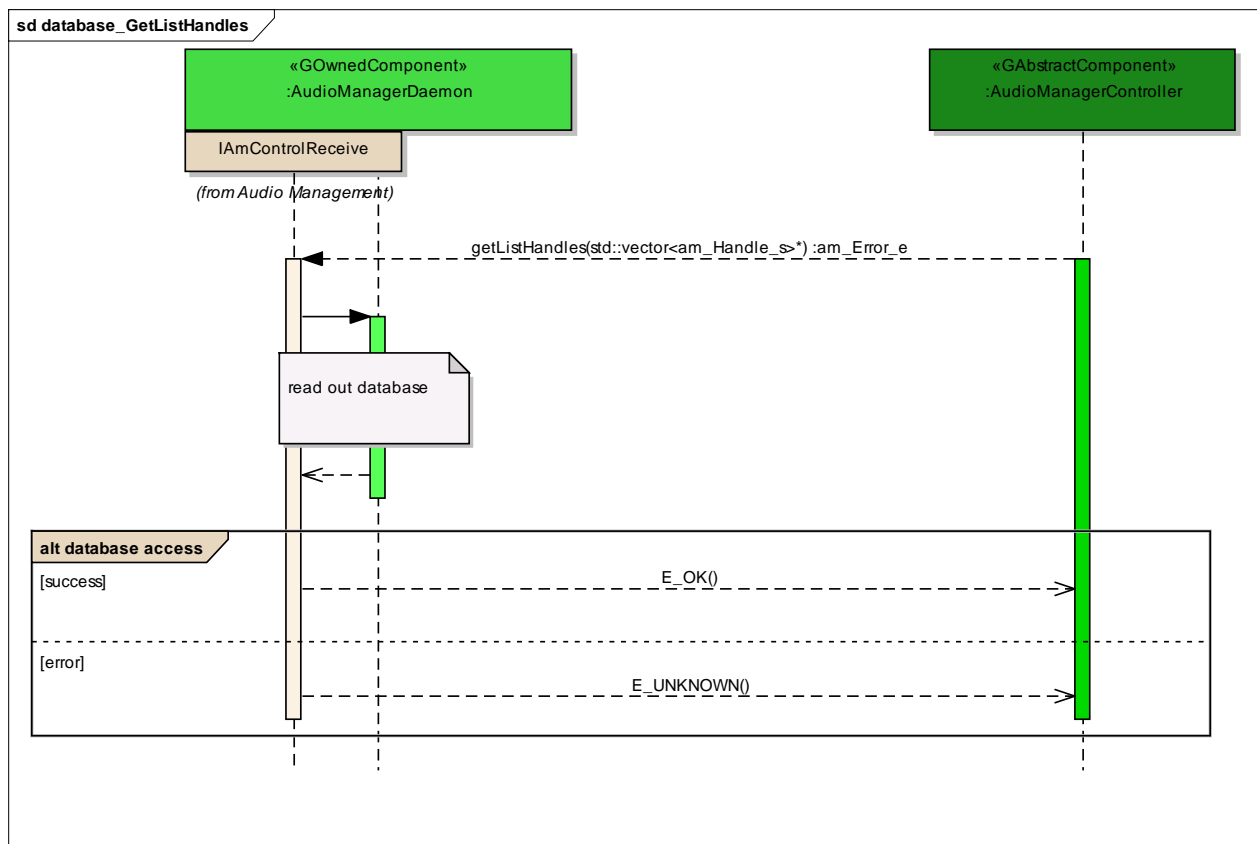
Figure: 31

database access

Type: **InteractionFragment**

Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListHandles *Keywords:*
Detail:	*Created on 07.10.2011. Last modified on 07.10.2011.*
GUID:	{62C0DDD7-4235-49f6-9EEA-0AC82EAD67C7}

database_GetListSinkClasses

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 23.09.2011. Last modified on 07.10.2011*
GUID:	{561CB8AA-535F-453f-81FC-775243C0F2FE}

database_GetListSinkClasses - *(Interaction diagram)*

Created By:	christian *on* 23.09.2011
Last Modified:	07.10.2011
Version:	1.0. *Locked:* False
GUID:	{77E4F69B-6F75-47ca-AEB1-3FB98F4BB0AB}
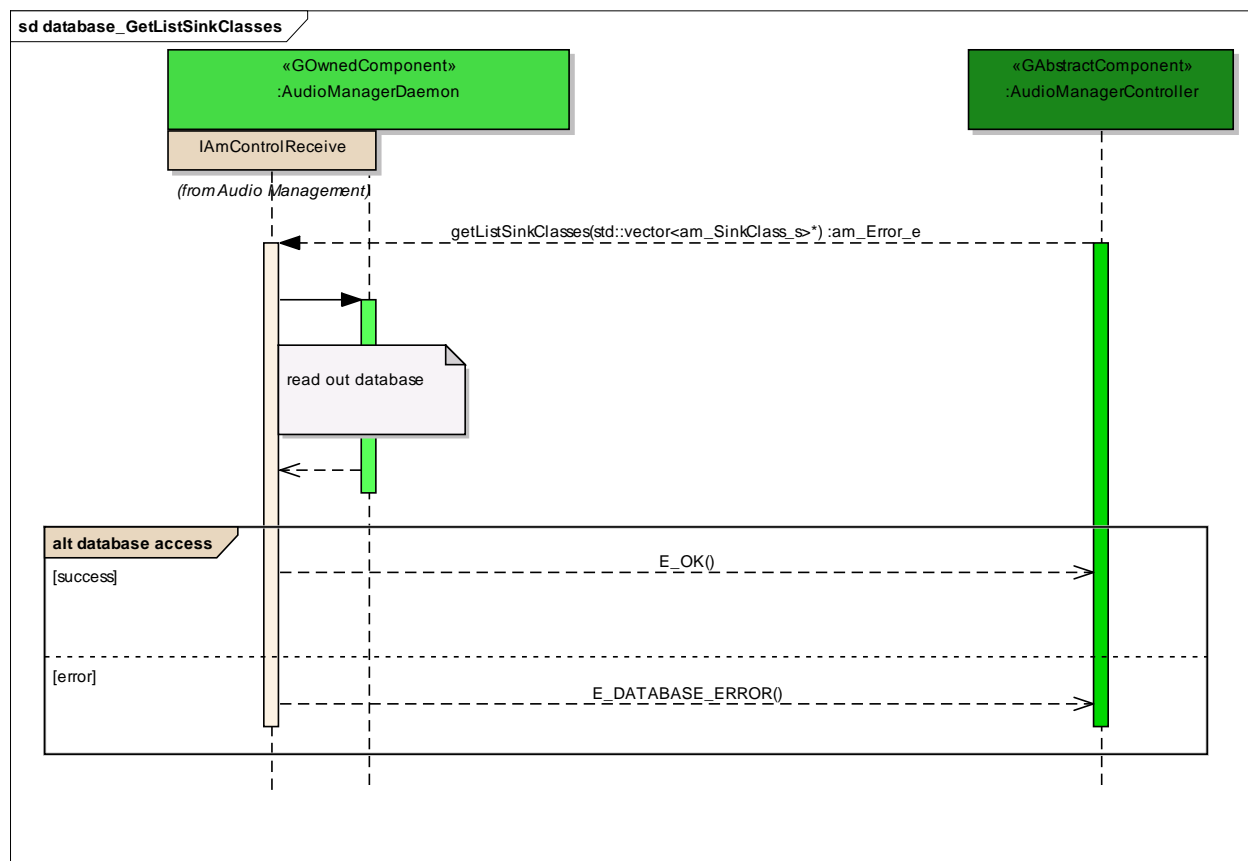
Figure: 32

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListSinkClasses *Keywords:*
Detail:	*Created on 07.10.2011. Last modified on 07.10.2011.*
GUID:	{53AC7085-87F5-4780-8F40-A78A1533CAA2}

database_GetListSourceClasses

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 23.09.2011. Last modified on 07.10.2011*
GUID:	{2A1A6397-9FD3-46ec-A4B3-46774ACB9095}

database_GetListSourceClasses - *(Interaction diagram)*

Created By:	christian *on* 23.09.2011
Last Modified:	07.10.2011
Version:	1.0. *Locked:* False
GUID:	{C6FE8607-D66E-4700-B03F-0A2B0D1CC22C}

Figure: 33

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListSourceClasses *Keywords:*
Detail:	*Created on 07.10.2011. Last modified on 07.10.2011.*
GUID:	{75F8BCDA-C3A7-4d5e-92E1-961ED674CEC9}

database_GetListSources

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 23.09.2011. Last modified on 07.10.2011*
GUID:	{12FD5F2E-7ABA-4249-950A-C308C81ED46D}

database_GetListSources - *(Interaction diagram)*

Created By:	christian *on* 23.09.2011
Last Modified:	07.10.2011
Version:	1.0. *Locked:* False
GUID:	{6122DA83-5ED9-4f9a-A0F2-C687AECCB5D4}

Figure: 34

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListSources *Keywords:*
Detail:	*Created on 07.10.2011. Last modified on 07.10.2011.*
GUID:	{3099799B-A90A-4dcb-BD75-6F7E02B6D166}

database_GetListSourcesOfDomain

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{B3B55DC1-02F6-4129-8DB5-7142E81F710C}

database_GetListSourcesOfDomain - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{515D0299-5BC6-4e0d-9439-756B831AE259}
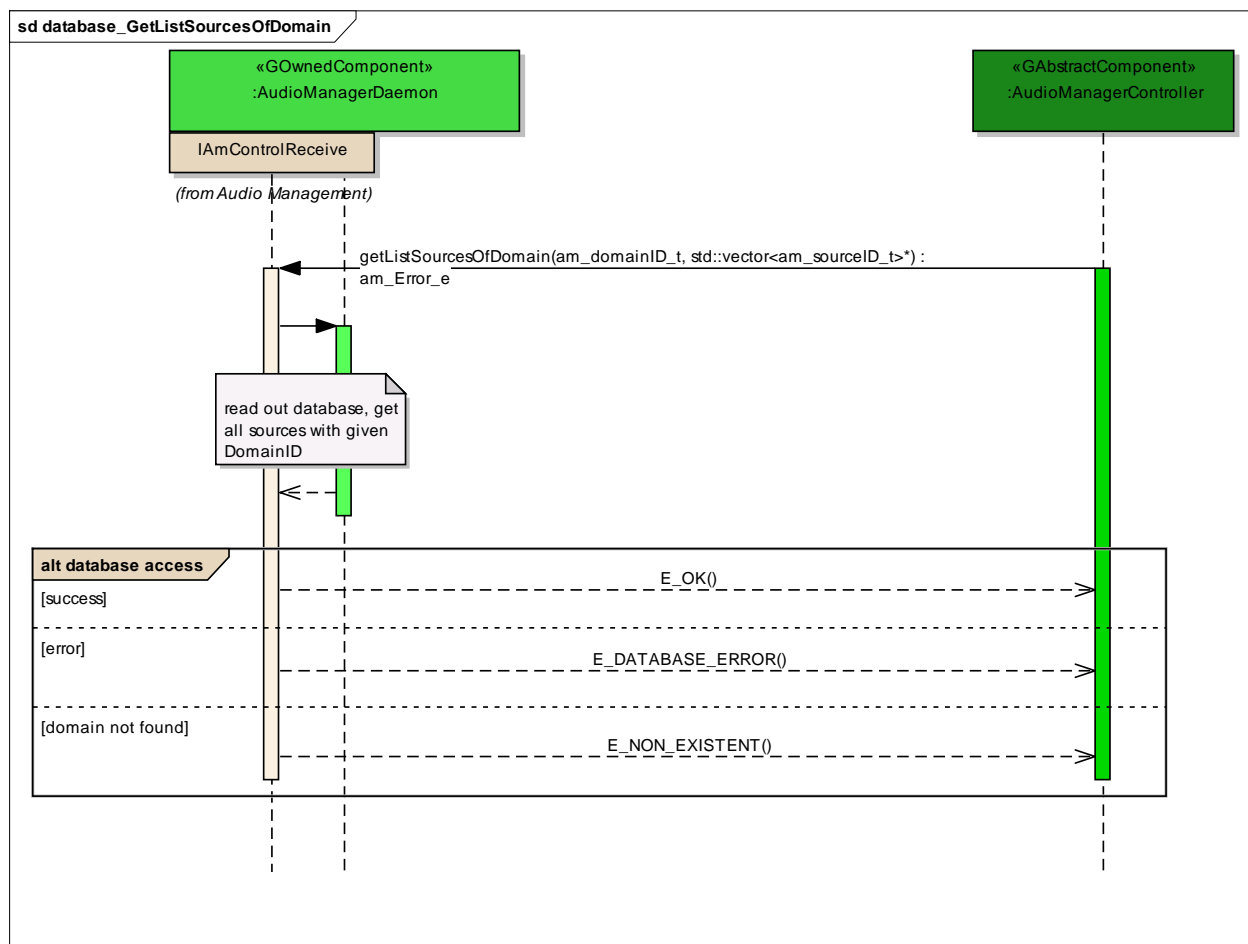
Figure: 35

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListSourcesOfDomain *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{854C4276-5FDA-4525-A704-4A77E00E06AB}

database_GetListSinks

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.

Package:	Database Access
Detail:	*Created on 23.09.2011. Last modified on 07.10.2011*
GUID:	{BF6DCA91-5D58-451e-ABAD-10CC8F2FD797}

database_GetListSinks - *(Interaction diagram)*

Created By:	christian *on* 23.09.2011
Last Modified:	07.10.2011
Version:	1.0. *Locked:* False
GUID:	{507D1EC3-673C-42c7-A42A-69AF90A42759}
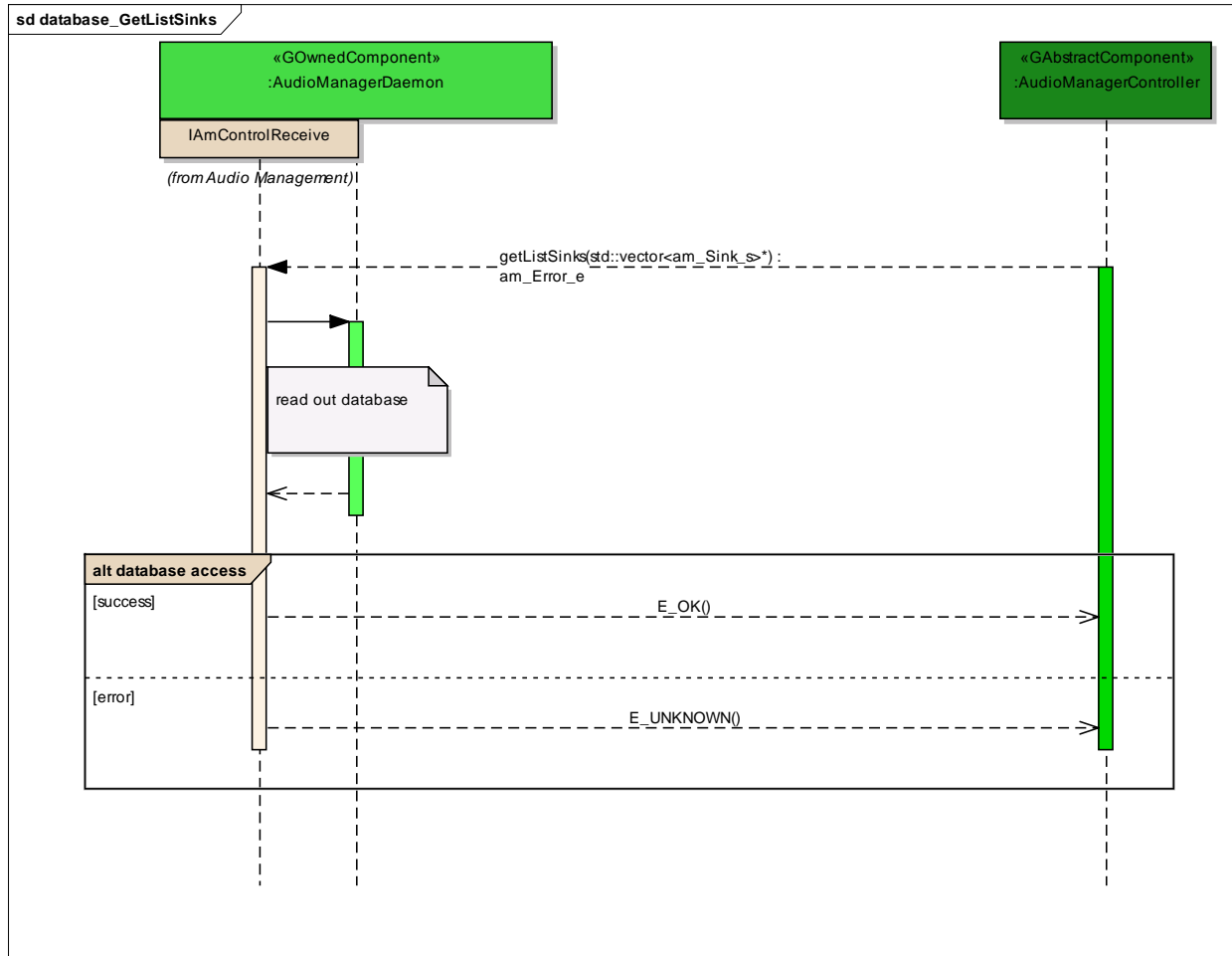
Figure: 36

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListSinks *Keywords:*
Detail:	*Created on 07.10.2011. Last modified on 07.10.2011.*

| *GUID:* | {C2BB35EA-255C-433a-8542-188B9E4AE88D} |

database_GetListSinksOfDomain

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{A723E159-A8C3-42af-BF5B-399F97A3CB2C}

database_GetListSinksOfDomain - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{FB624010-21A3-4081-BEF3-F4548042F183}
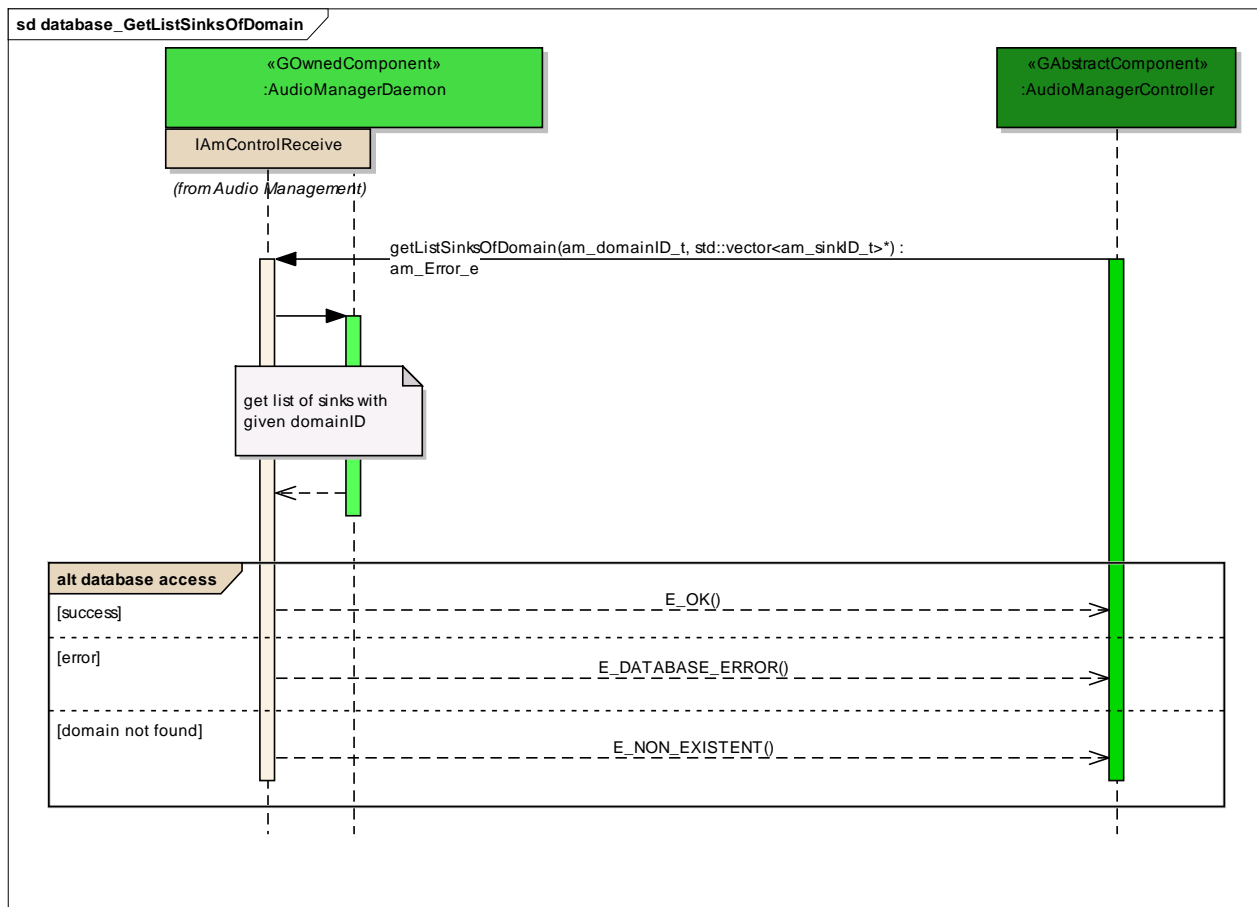
Figure: 37

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListSinksOfDomain *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{73F01233-775D-40f6-97B0-7D2FD9FEF9FF}

database_GetListConnections

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access

database_GetListConnections - *(Interaction diagram)*

Created By: christian *on* 23.09.2011
Last Modified: 07.10.2011
Version: 1.0. *Locked:* False
GUID: {2CBA0F94-3006-492d-8E8C-5886A19BD7AE}
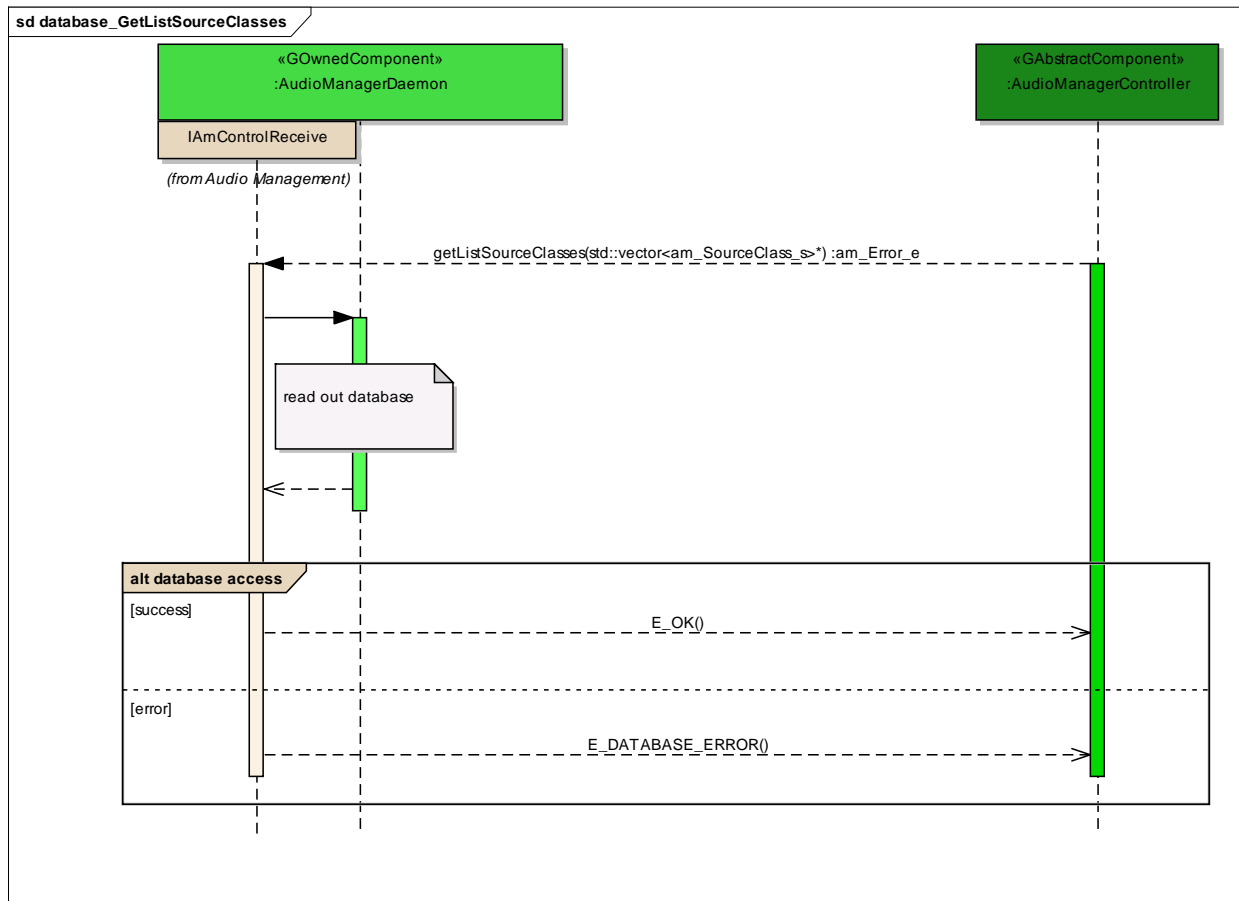

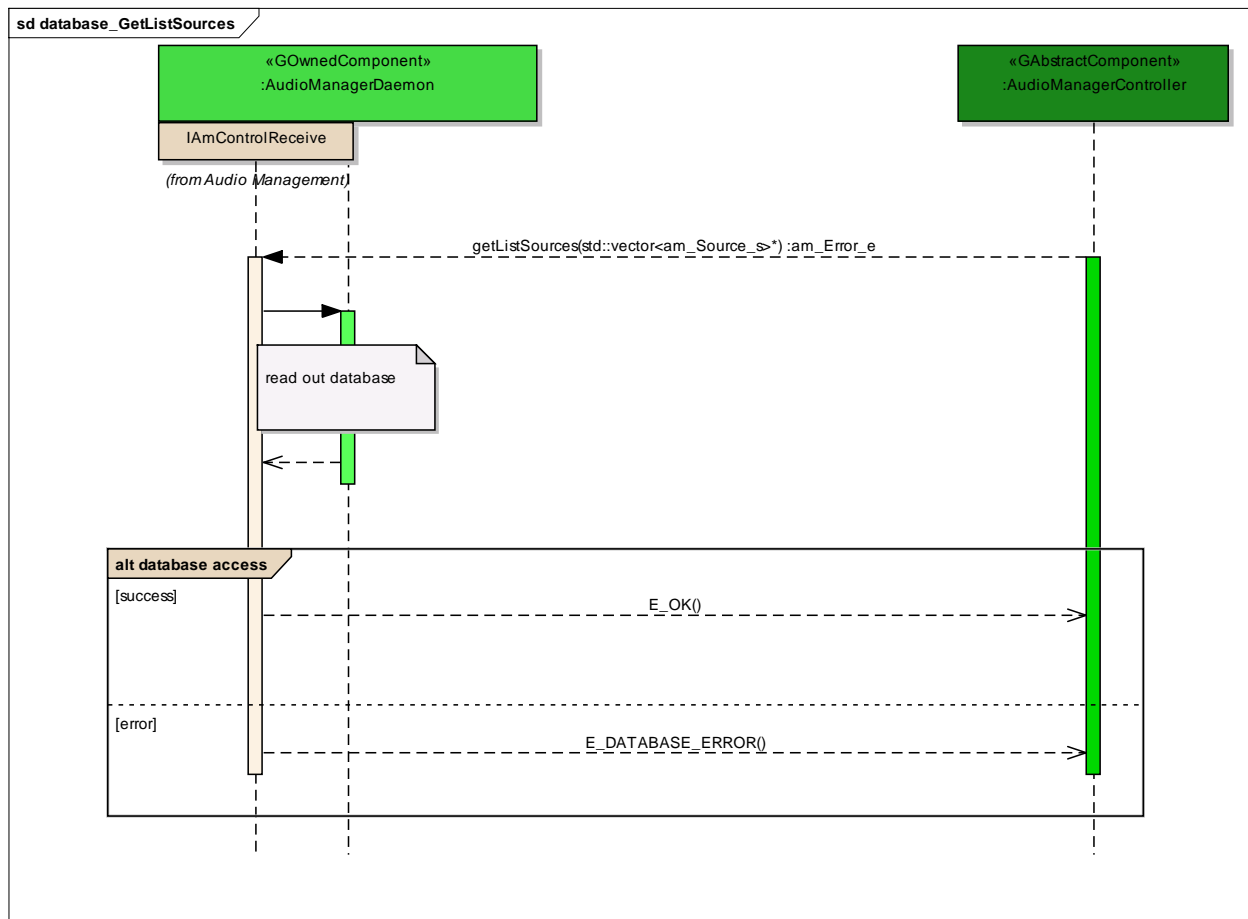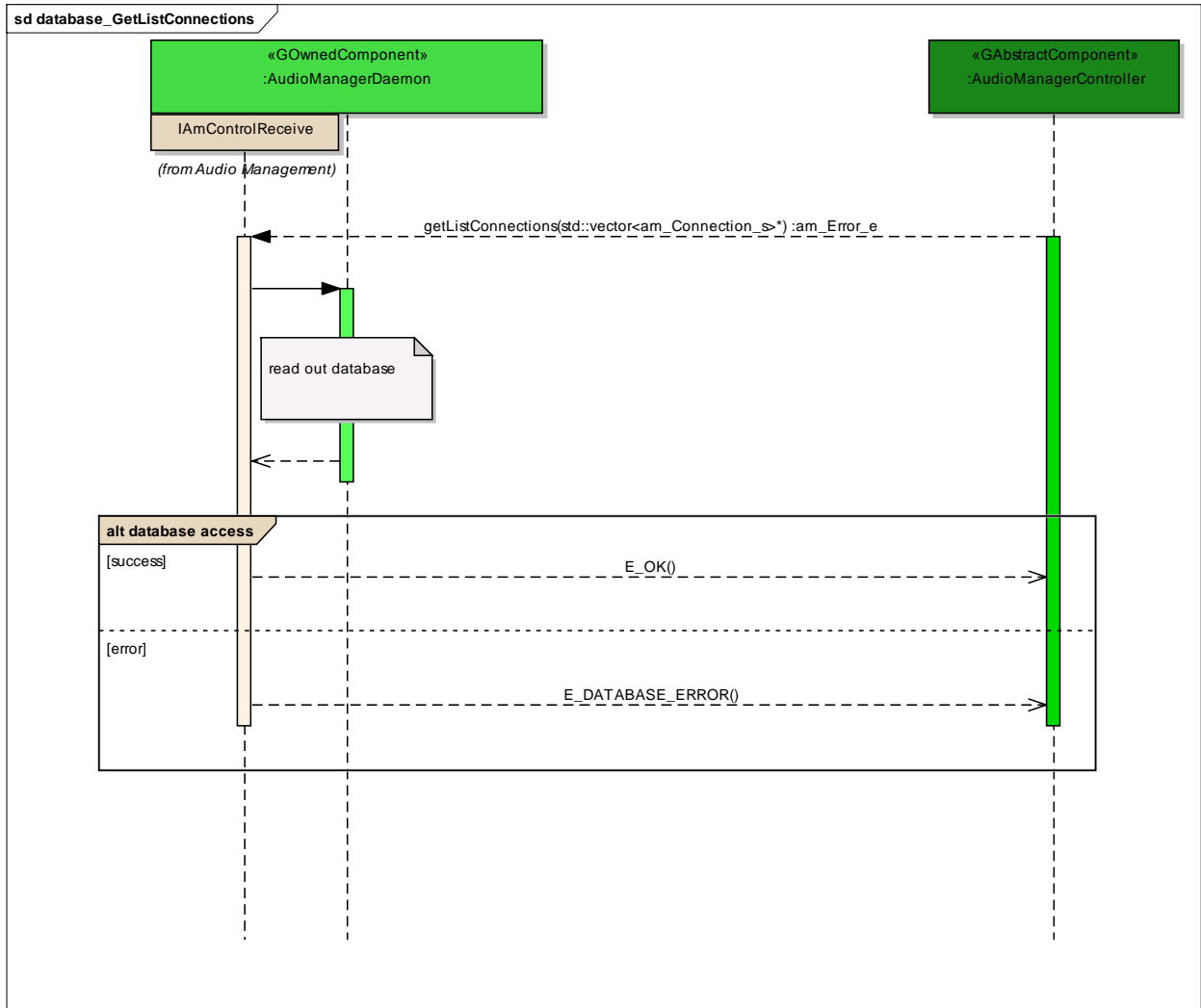Figure: 38

database access

Type: **InteractionFragment**
Status: Proposed. Version 1.0. Phase 1.0.

Package: database_GetListConnections   *Keywords:*
Detail: *Created on 07.10.2011. Last modified on 07.10.2011.*
GUID: {951B2B4D-6A3D-43dd-9713-9F029DEF3DD5}

database_GetListDomains

Type: **Package**
Status: Reworked. Version 1.0. Phase 1.0.
Package: Database Access
Detail: *Created on 23.09.2011. Last modified on 07.10.2011*
GUID: {ACE45F10-68F8-41f7-971F-CB7A0F7E6ECE}

database_GetListDomains - *(Interaction diagram)*

Created By: christian *on* 23.09.2011
Last Modified: 07.10.2011
Version: 1.0. *Locked:* False
GUID: {32321FD4-B506-42c9-9435-9F444FF5EB8B}
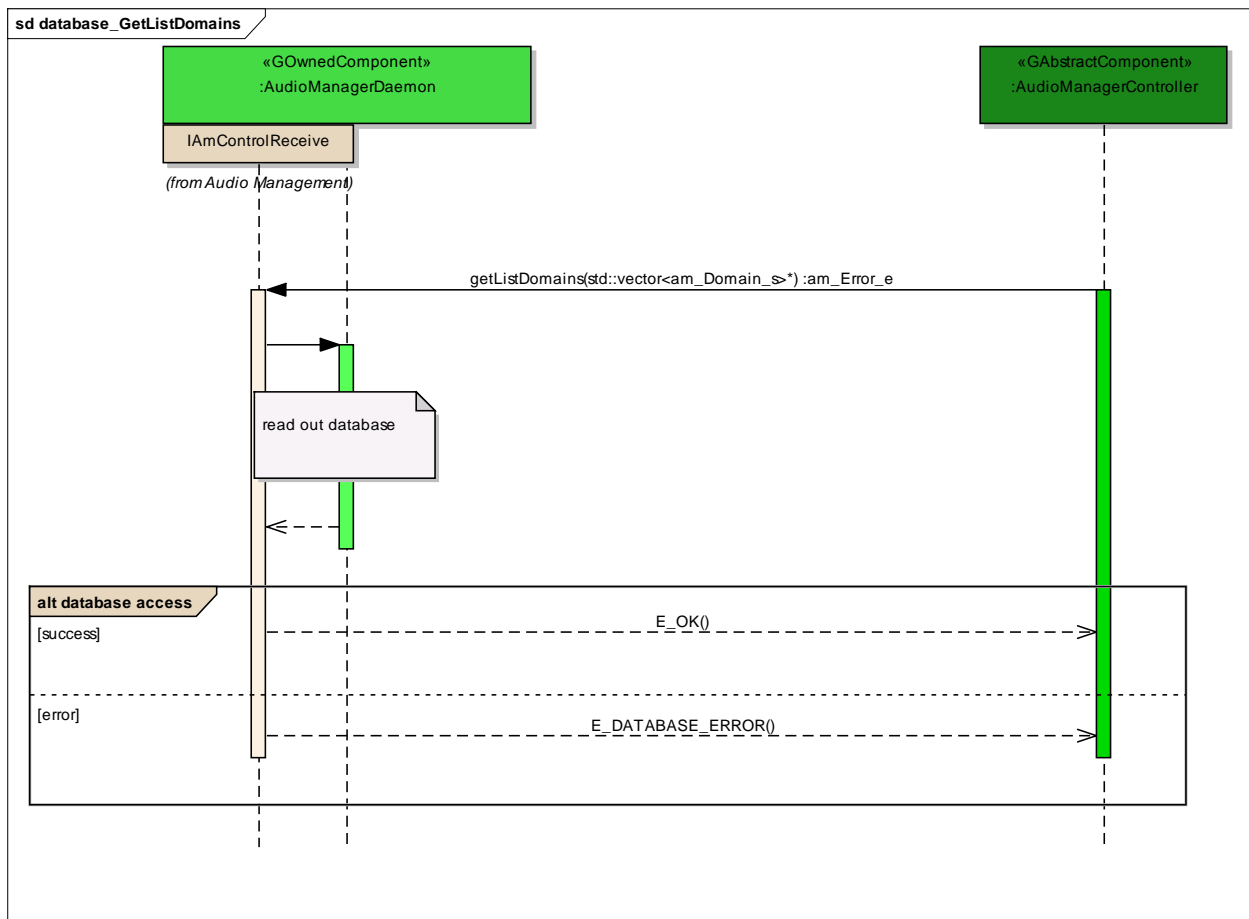
Figure: 39

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListDomains *Keywords:*
Detail:	*Created on 07.10.2011. Last modified on 07.10.2011.*
GUID:	{09793094-9B3E-48b6-9792-328AA688E32F}

database_GetListMainConnections

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access

database_GetListMainConnections - *(Interaction diagram)*
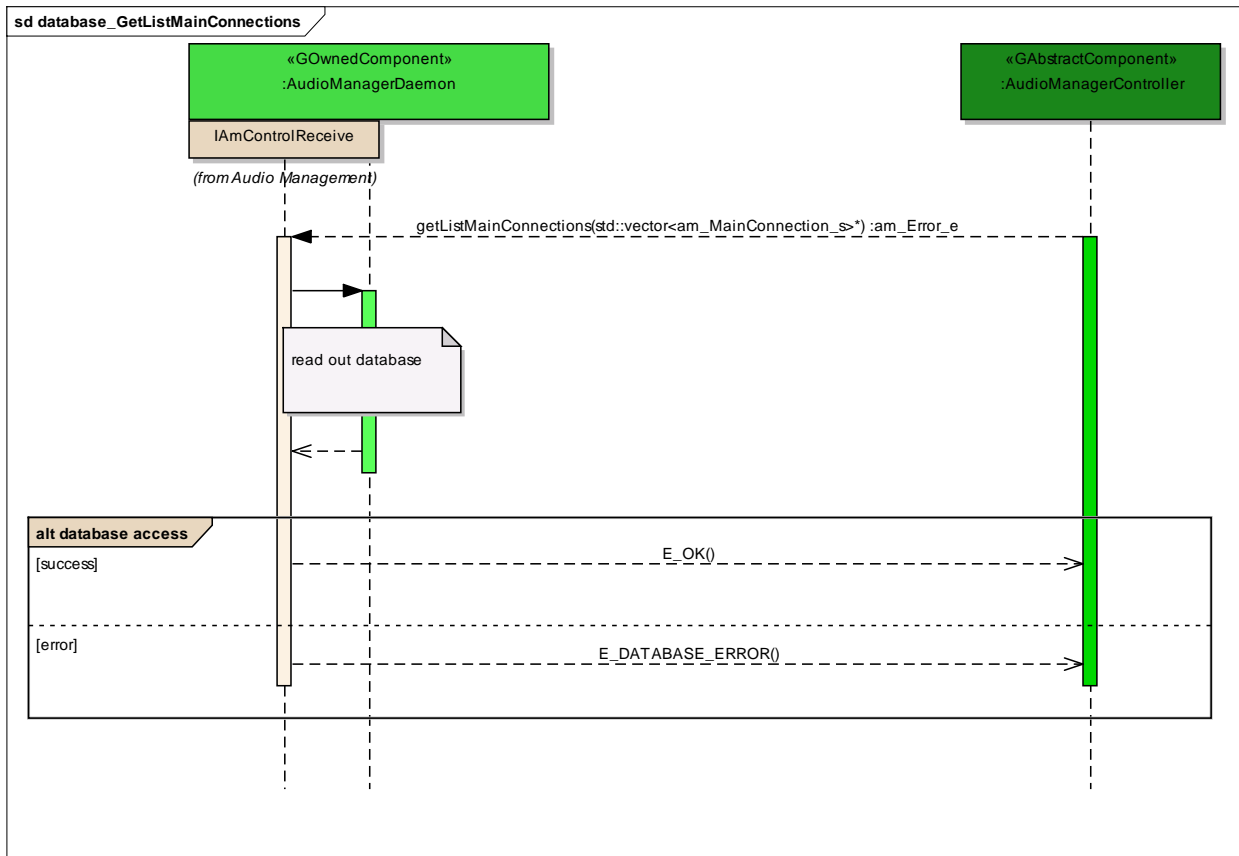
Figure: 40

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListMainConnections *Keywords:*
Detail:	*Created on 07.10.2011. Last modified on 07.10.2011.*
GUID:	{A542CF87-612B-4e45-BEC2-448F9A402551}

database_GetListSystemProperties

Type: **Package**
Status: Proposed. Version 1.0. Phase 1.0.
Package: Database Access
Detail: *Created on 07.12.2011. Last modified on 07.12.2011*
GUID: {3E8BD4CB-AE72-48b5-A1BE-6B2AD3509D30}

database_GetListSystemProperties - *(Interaction diagram)*

Created By: christian *on* 07.12.2011
Last Modified: 07.12.2011
Version: 1.0. *Locked:* False
GUID: {4F96A831-DF1A-450c-9CEB-D4AF333C27E7}
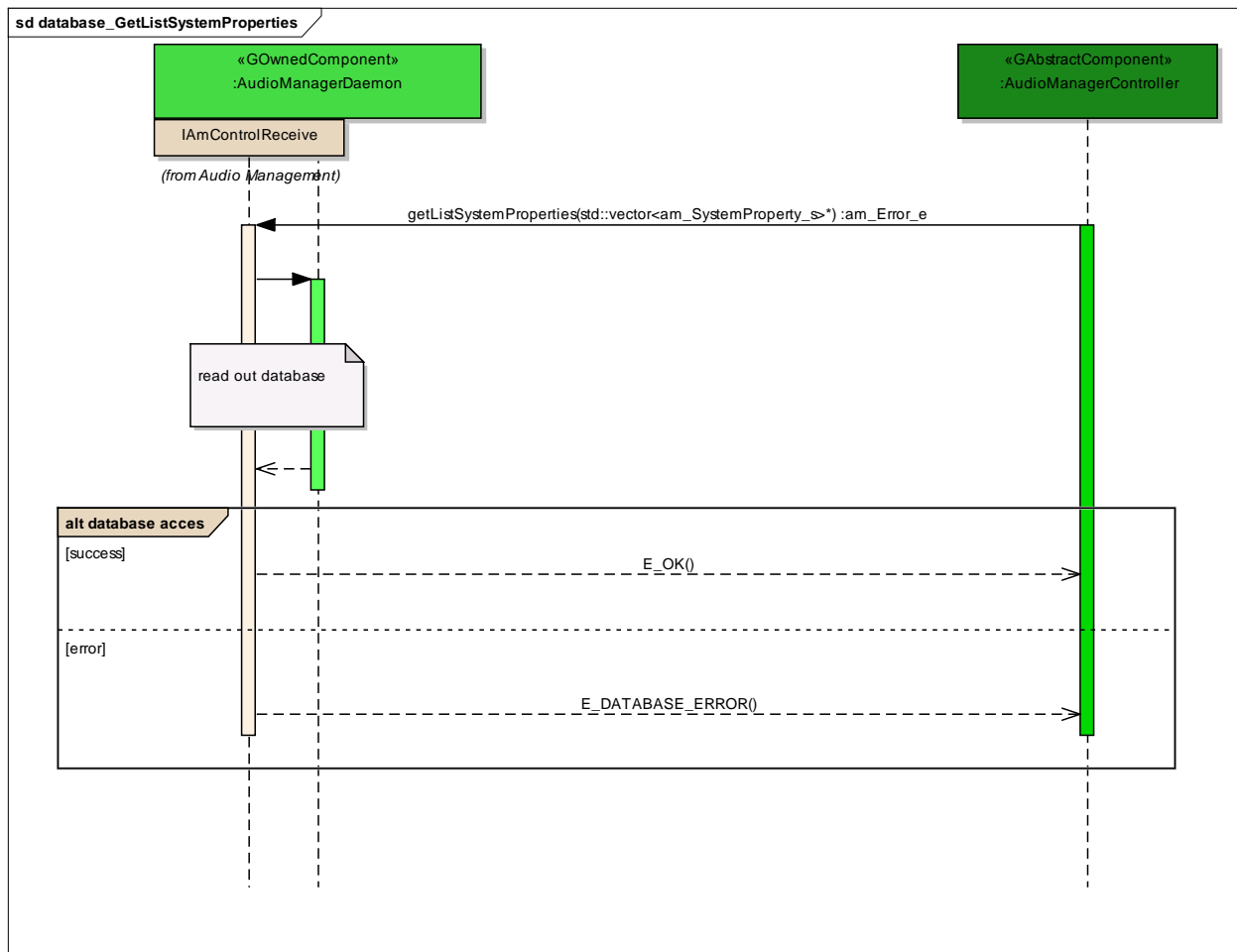
Figure: 41

database acces

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetListSystemProperties *Keywords:*
Detail:	*Created on 07.12.2011. Last modified on 07.12.2011.*
GUID:	{4E83FD7B-289B-49cd-8A06-833C0B401C7C}

database_GetSinkClass

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 14.09.2011. Last modified on 07.10.2011*
GUID:	{A249DF43-4A01-4c08-8A59-39A1FCA62370}

database_GetSinkClass - *(Interaction diagram)*

Created By:	christian *on* 14.09.2011
Last Modified:	07.10.2011
Version:	1.0. *Locked:* False
GUID:	{C8687129-D66F-47ec-BA00-F100E96FA067}
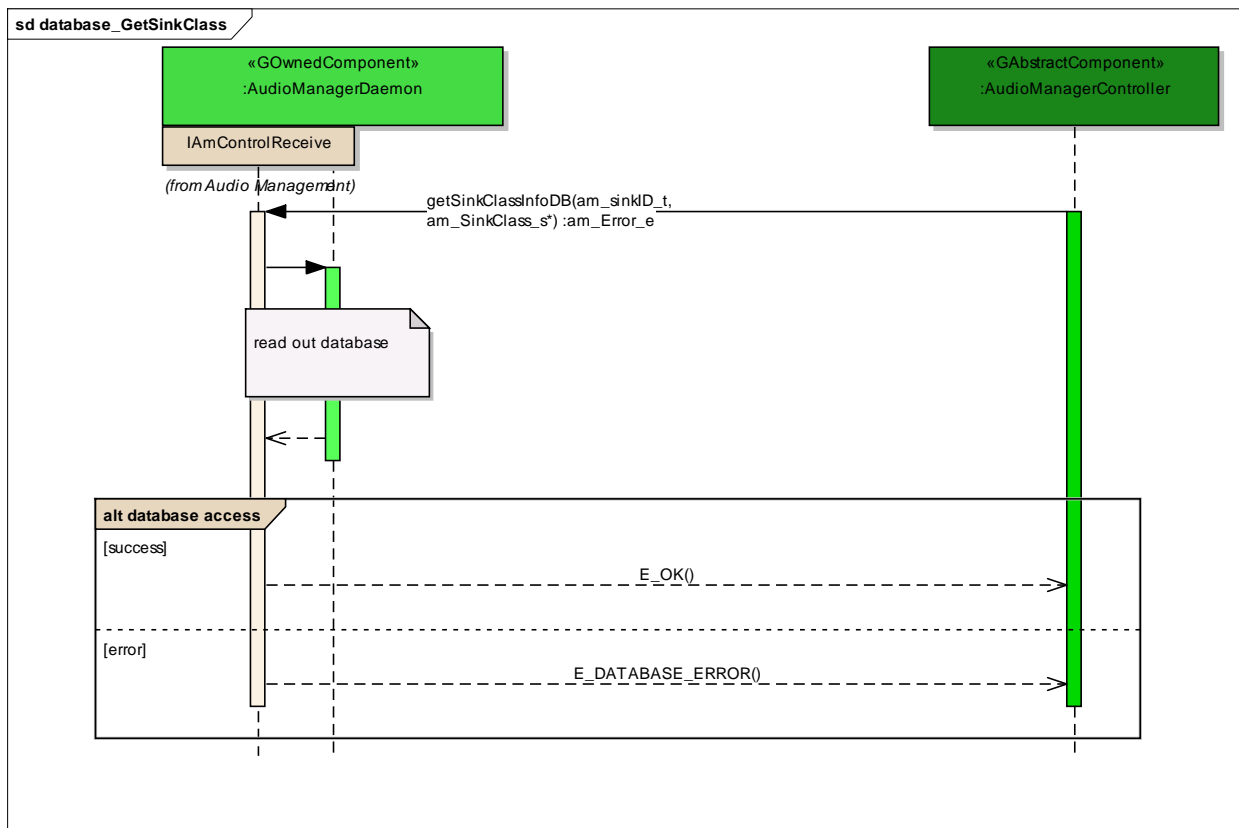
Figure: 42

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetSinkClass *Keywords:*
Detail:	*Created on 07.10.2011. Last modified on 07.10.2011.*
GUID:	{27A53204-424E-467c-B4F2-04C4F1C0CBA7}

database_GetSourceClass

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 13.09.2011. Last modified on 07.10.2011*
GUID:	{D055786C-51CB-4558-8471-75D3BE9F60D0}

database_GetSourceClass - *(Interaction diagram)*

Created By:	christian *on* 13.09.2011
Last Modified:	07.10.2011
Version:	1.0. *Locked:* False
GUID:	{0A76F246-18A3-43c7-8FC4-BC811C183AA8}
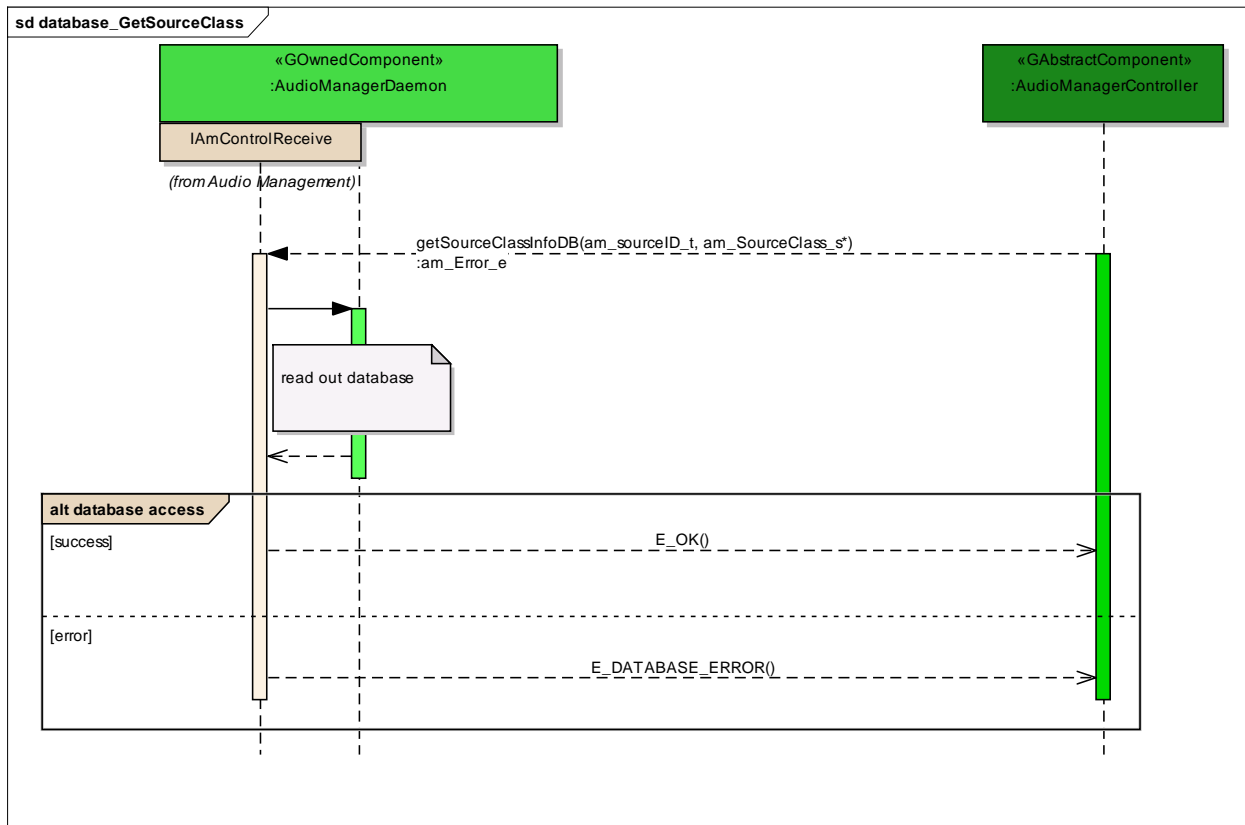
Figure: 43

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_GetSourceClass *Keywords:*
Detail:	*Created on 07.10.2011. Last modified on 07.10.2011.*
GUID:	{2325A660-655E-4ce0-9BAE-1B4E5C1DD045}

database_ChangeSinkClass

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 15.09.2011. Last modified on 08.12.2011*
GUID:	{93A31C05-37BB-42bf-BB90-24A3998A209B}

database_ChangeSinkClass - *(Interaction diagram)*

Created By:	christian *on* 15.09.2011
Last Modified:	08.12.2011
Version:	1.0. *Locked:* False
GUID:	{3FFC7646-8490-4a1c-B4BD-71565EA85BFC}
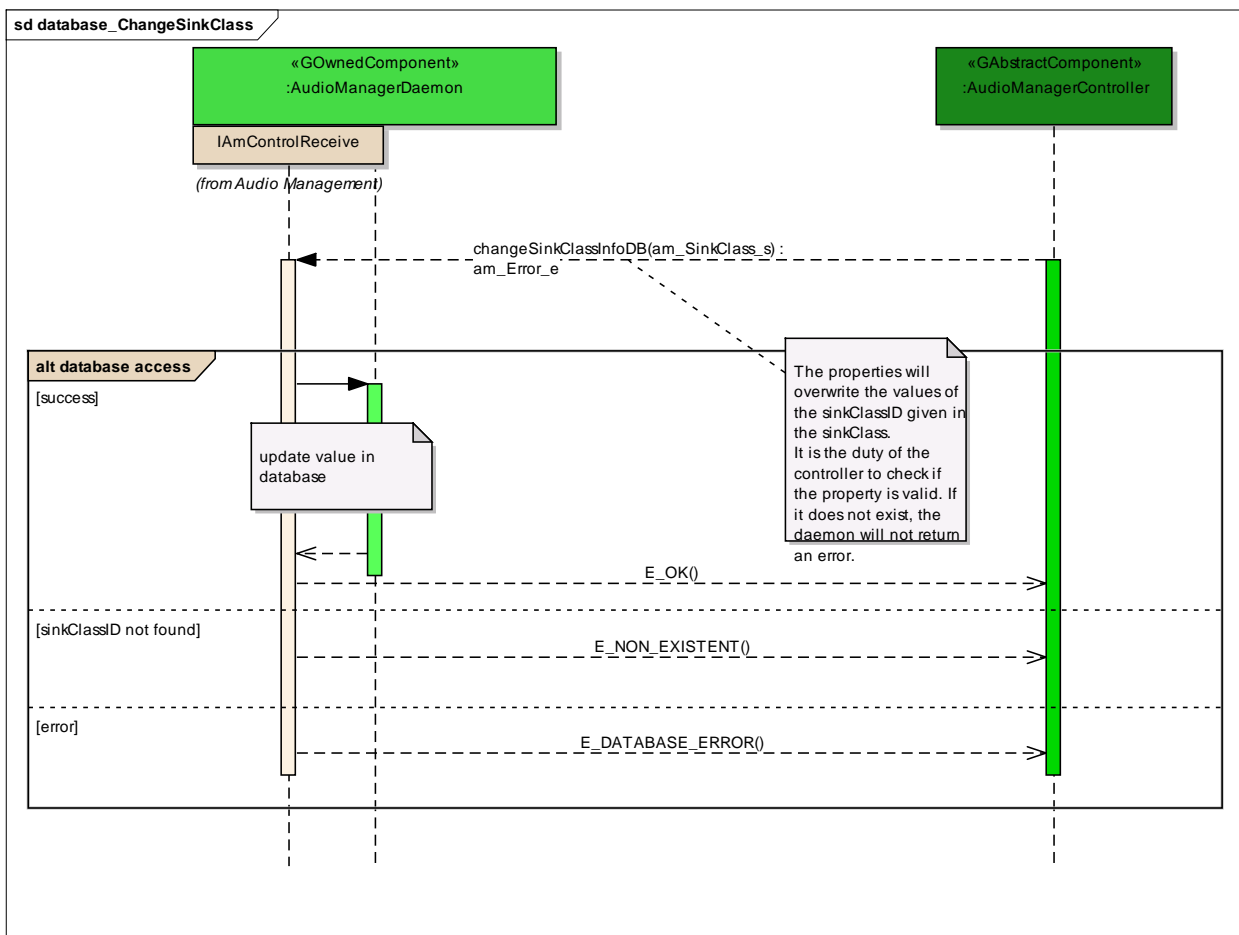
Figure: 44

database access

Type:	**InteractionFragment**

Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeSinkClass *Keywords:*
Detail:	*Created on 15.09.2011. Last modified on 08.12.2011.*
GUID:	{1986FC88-1B1E-4805-A142-768F8C127760}

database_ChangeSourceClass

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 13.09.2011. Last modified on 08.12.2011*
GUID:	{DFF064FC-4C8B-4e85-B219-3A1BE890660D}

database_ChangeSourceClass - *(Interaction diagram)*

Created By:	christian *on* 13.09.2011
Last Modified:	08.12.2011
Version:	1.0. *Locked:* False
GUID:	{8E132AAE-EC41-43a3-90D9-6E66FA10B288}
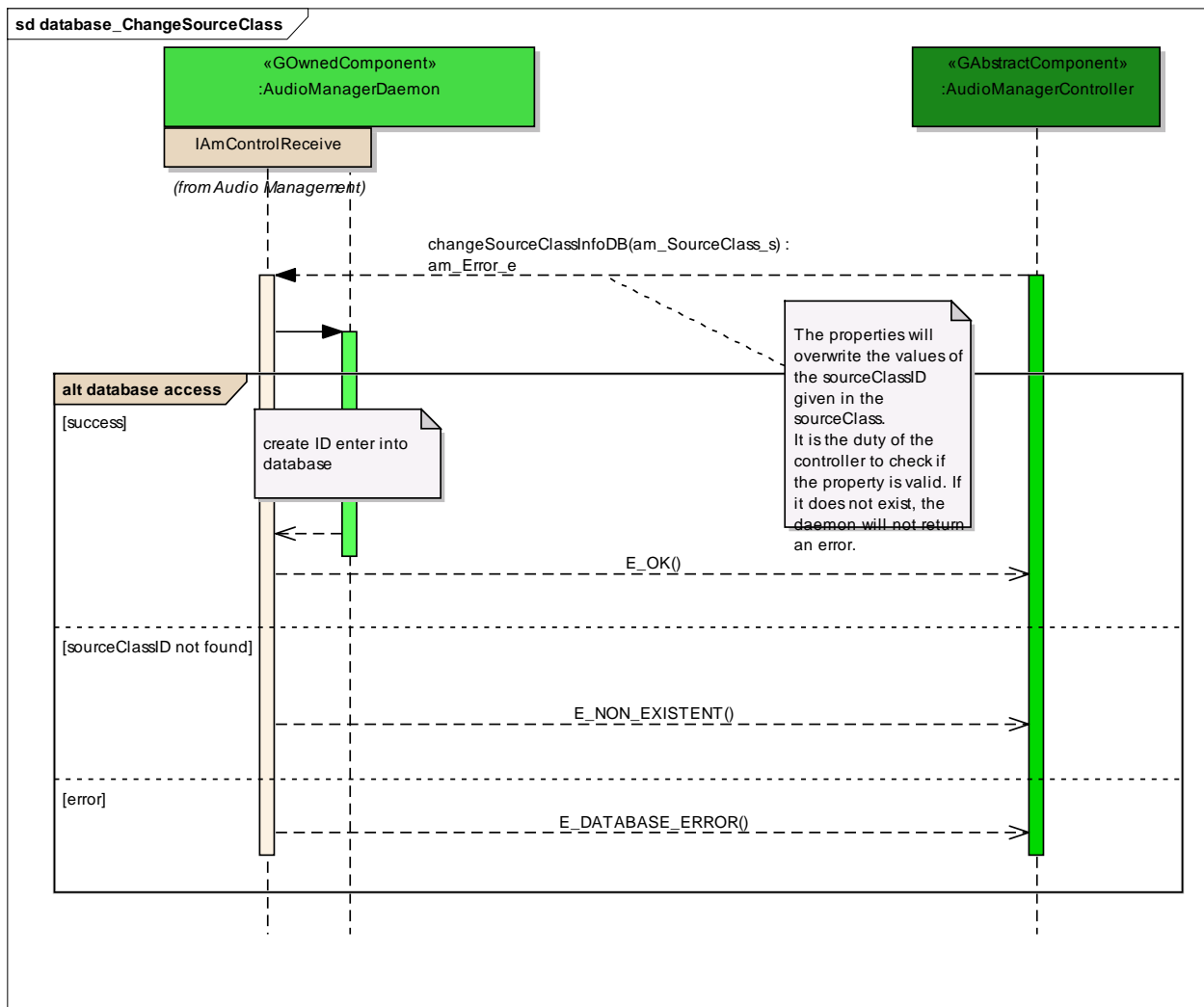
Figure: 45

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeSourceClass *Keywords:*
Detail:	*Created on 14.09.2011. Last modified on 08.12.2011.*
GUID:	{37CE53CF-F97B-454d-BE43-C09C9045059D}

database_ChangeMainConnectionState

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{852956B0-650A-46d8-847E-86BE47F4721D}

database_ChangeMainConnectionState - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{02BD03F9-0136-4c2e-B595-63FA33D7C4C9}
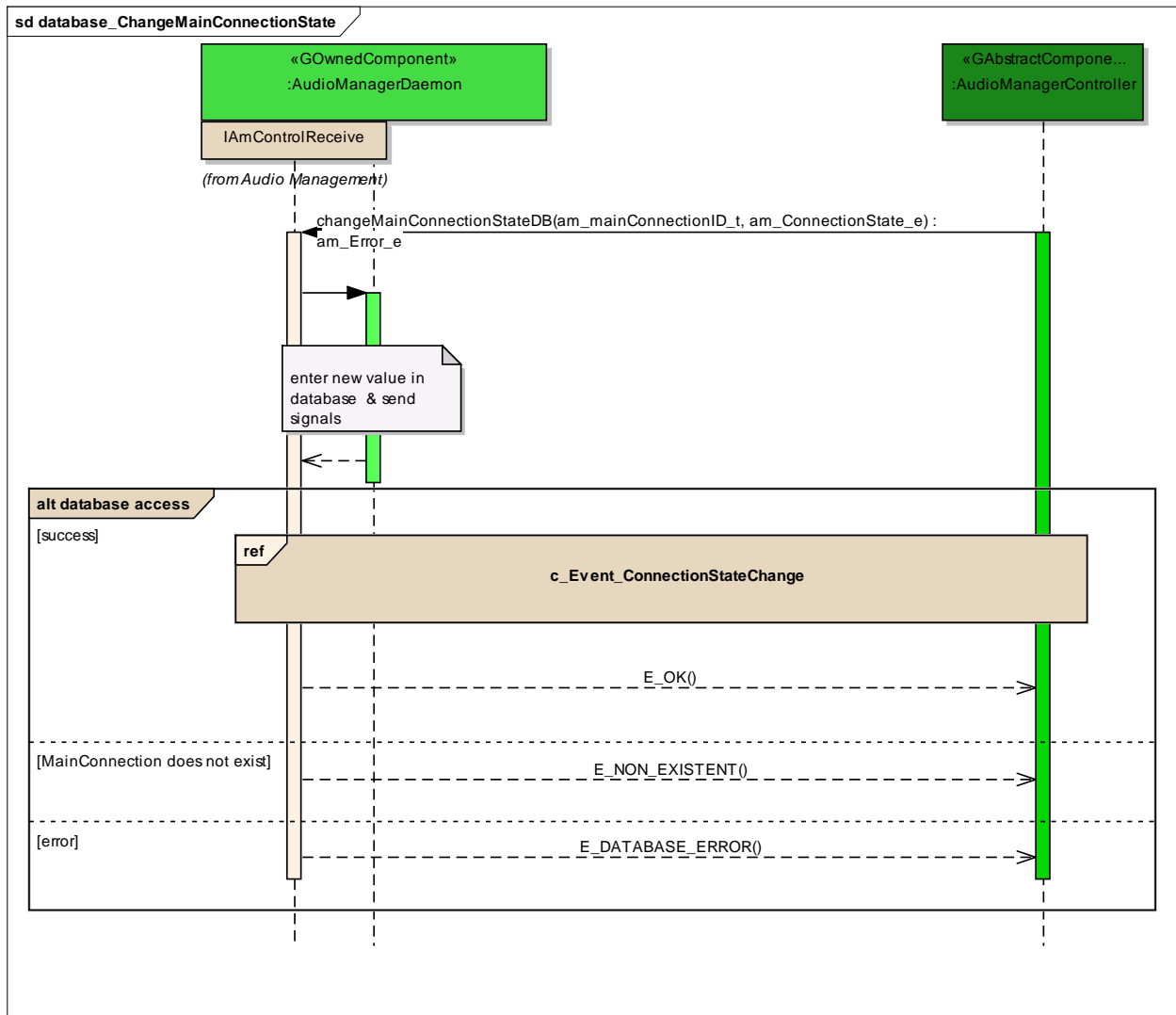
Figure: 46

c_Event_ConnectionStateChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMainConnectionState *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{BCB8206F-385F-472e-876C-3F96810620F7}

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMainConnectionState *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{89CB5CBC-988A-425f-BD6C-0E867CE01805}

database_ChangeMainConnectionRoute

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 20.10.2011. Last modified on 20.10.2011*
GUID:	{98D1AA91-1AC2-4ba1-B16E-D90CDFA8AD8F}

database_ChangeMainConnectionRoute - *(Interaction diagram)*

Created By:	christian *on* 20.10.2011
Last Modified:	20.10.2011
Version:	1.0. *Locked:* False
GUID:	{F14EEE14-1372-4c5b-862E-46E227B9843C}
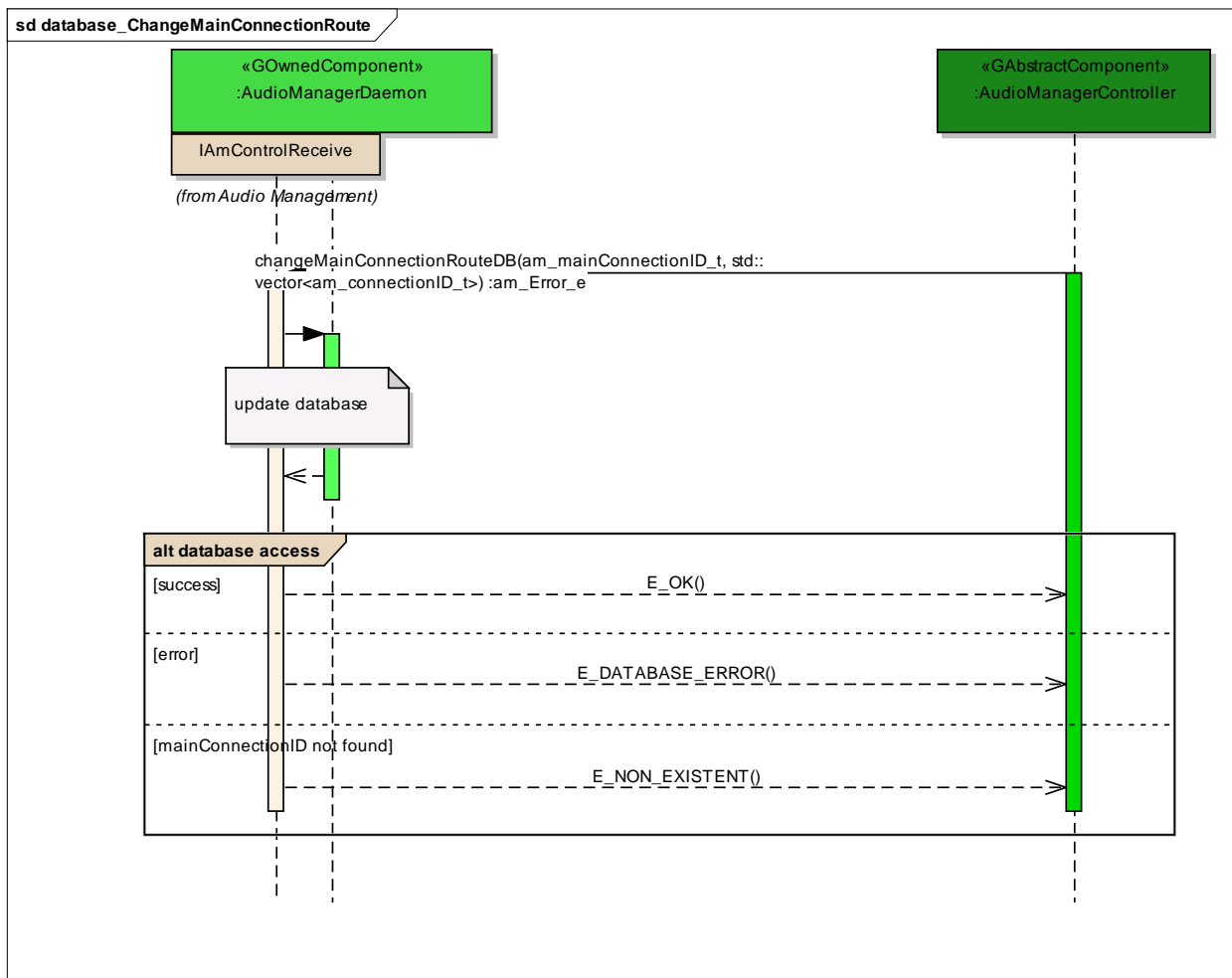
Figure: 47

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMainConnectionRoute *Keywords:*
Detail:	*Created on 20.10.2011. Last modified on 20.10.2011.*
GUID:	{783BFB7A-83AA-4e70-A46D-095A11790B9F}

database_ChangeMuteState

Type:	**Package**

Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{243AD1E8-E4F1-4b27-870C-B4A5280A166E}

database_ChangeMuteState - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{8474EC2B-2CEA-4de5-8227-F32C60E20CB9}
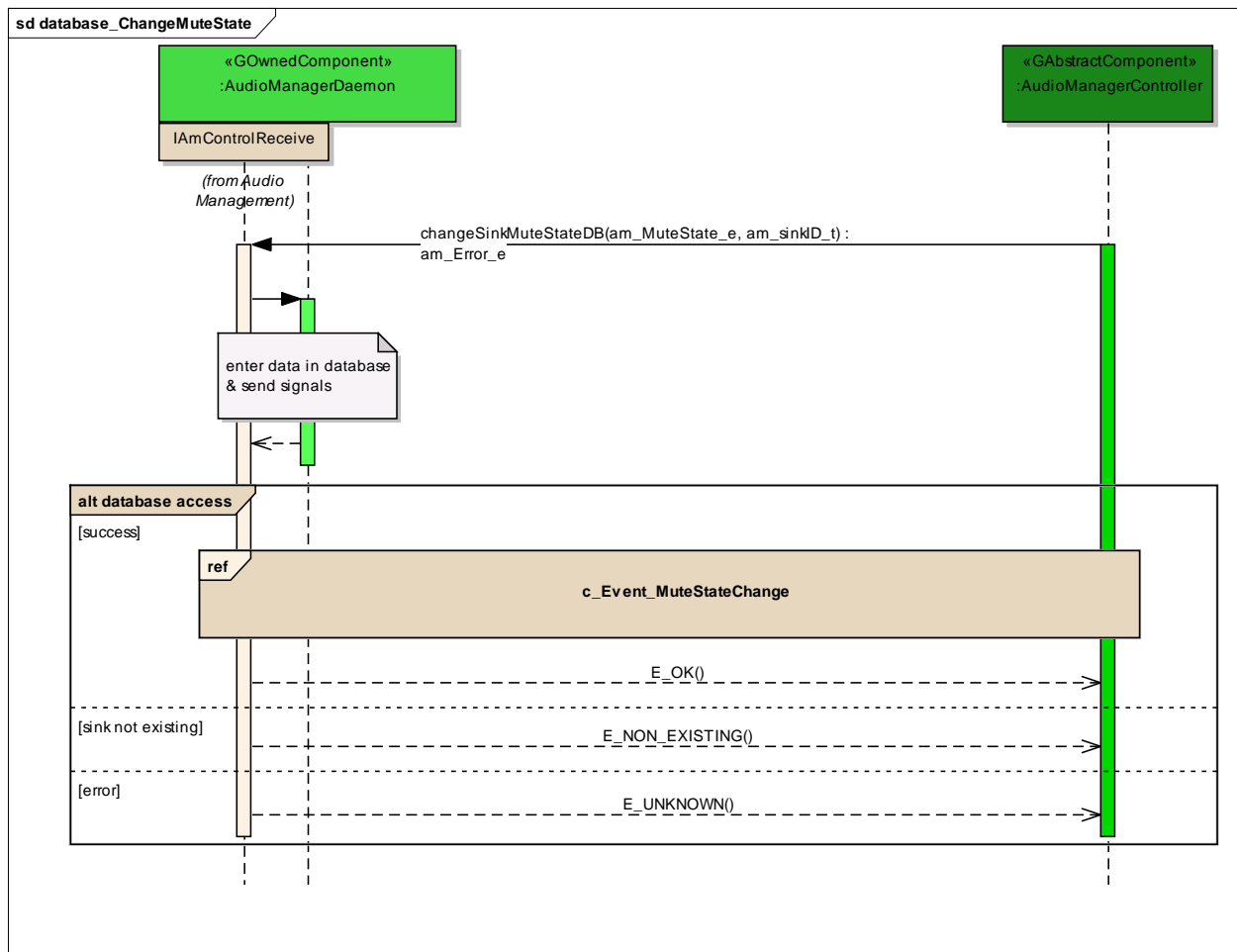
Figure: 48

c_Event_MuteStateChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMuteState *Keywords:*

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMuteState *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{8AFE6226-8195-4859-9398-EF0B65D36FD6}

database_ChangeMainSinkSoundProperty

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{F5AD364F-BE93-4990-B485-530DFE80217A}

database_ChangeMainSinkSoundProperty - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	08.12.2011
Version:	1.0. *Locked:* False
GUID:	{720A8D3A-CCBF-4f06-8382-701712D06ECD}
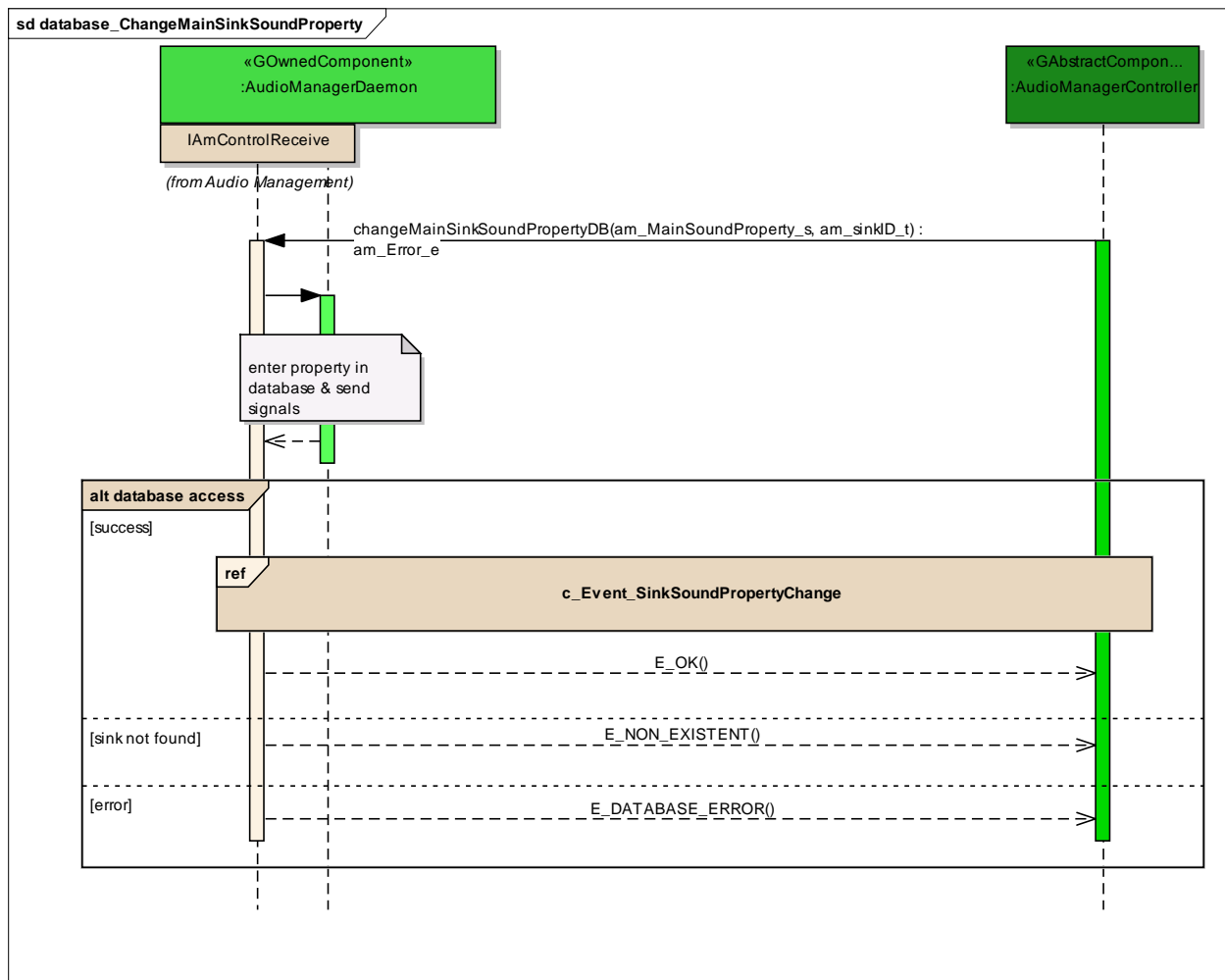
Figure: 49

c_Event_SinkSoundPropertyChange

Type:	__InteractionOccurrence__
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMainSinkSoundProperty *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{29D5D319-176C-40ec-87B6-638B3C8ED075}

database access

Type:	__InteractionFragment__
Status:	Proposed. Version 1.0. Phase 1.0.

Package:	database_ChangeMainSinkSoundProperty	*Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*	
GUID:	{60281A4D-AB89-4b84-9D0E-6EBCD2B35893}	

database_ChangeMainSourceSoundProperty

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{BB94662E-7BE9-4513-8407-33DF03760484}

database_ChangeMainSourceSoundProperty - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{94F22D32-CBAC-4bb4-9981-0CFED0A28230}
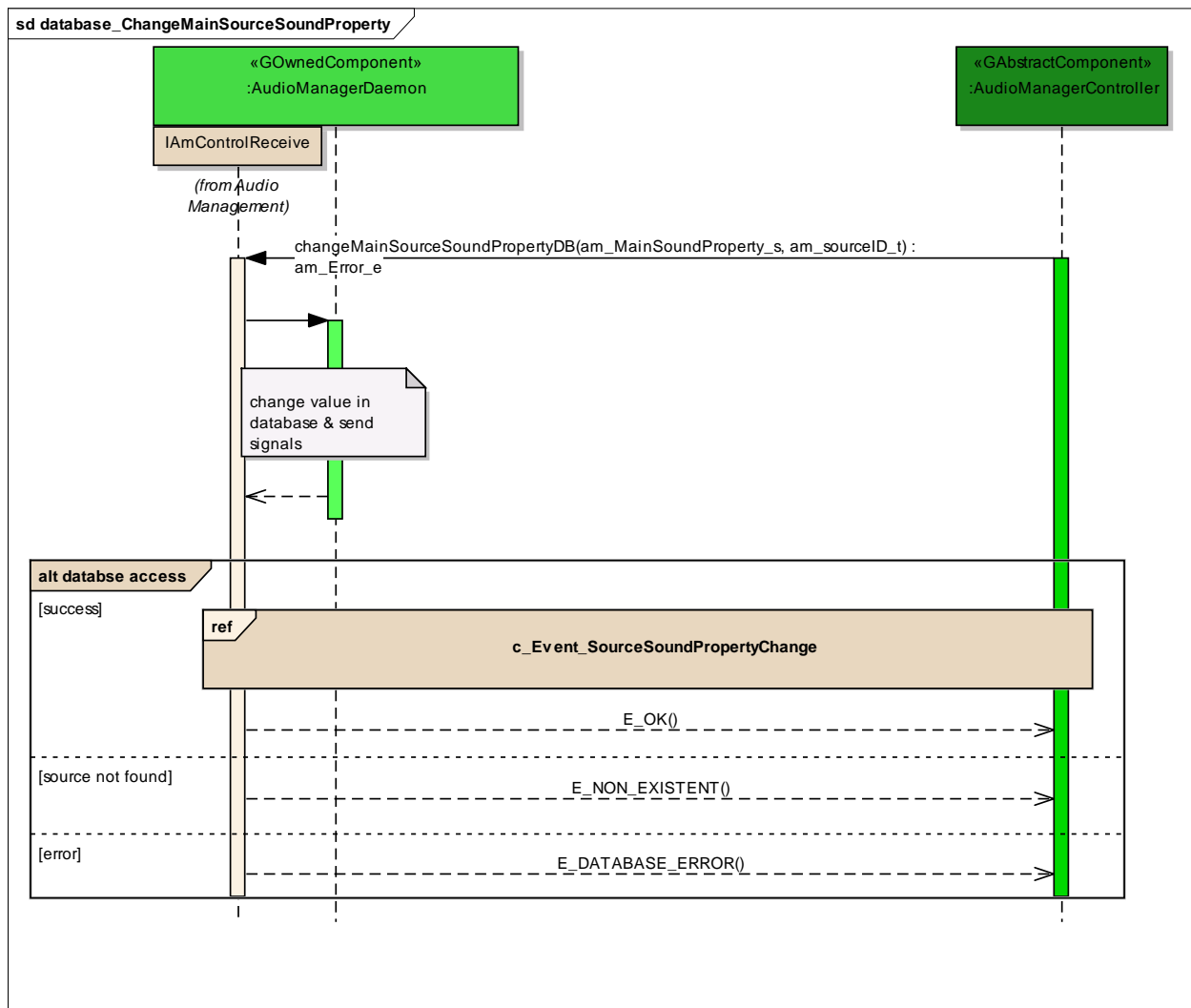
Figure: 50

c_Event_SourceSoundPropertyChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMainSourceSoundProperty *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{1EE682CB-144B-412f-A053-331842F3A8CE}

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMainSourceSoundProperty *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{CE49F5E6-CB3B-4062-8E9B-40898F4E4E78}

database_ChangeSystemProperty

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{19F022C9-3E53-4612-A2F9-A0733E23ADBF}

database_ChangeSystemProperty - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{9A8CDAE6-7588-464d-BADE-7E479F378169}
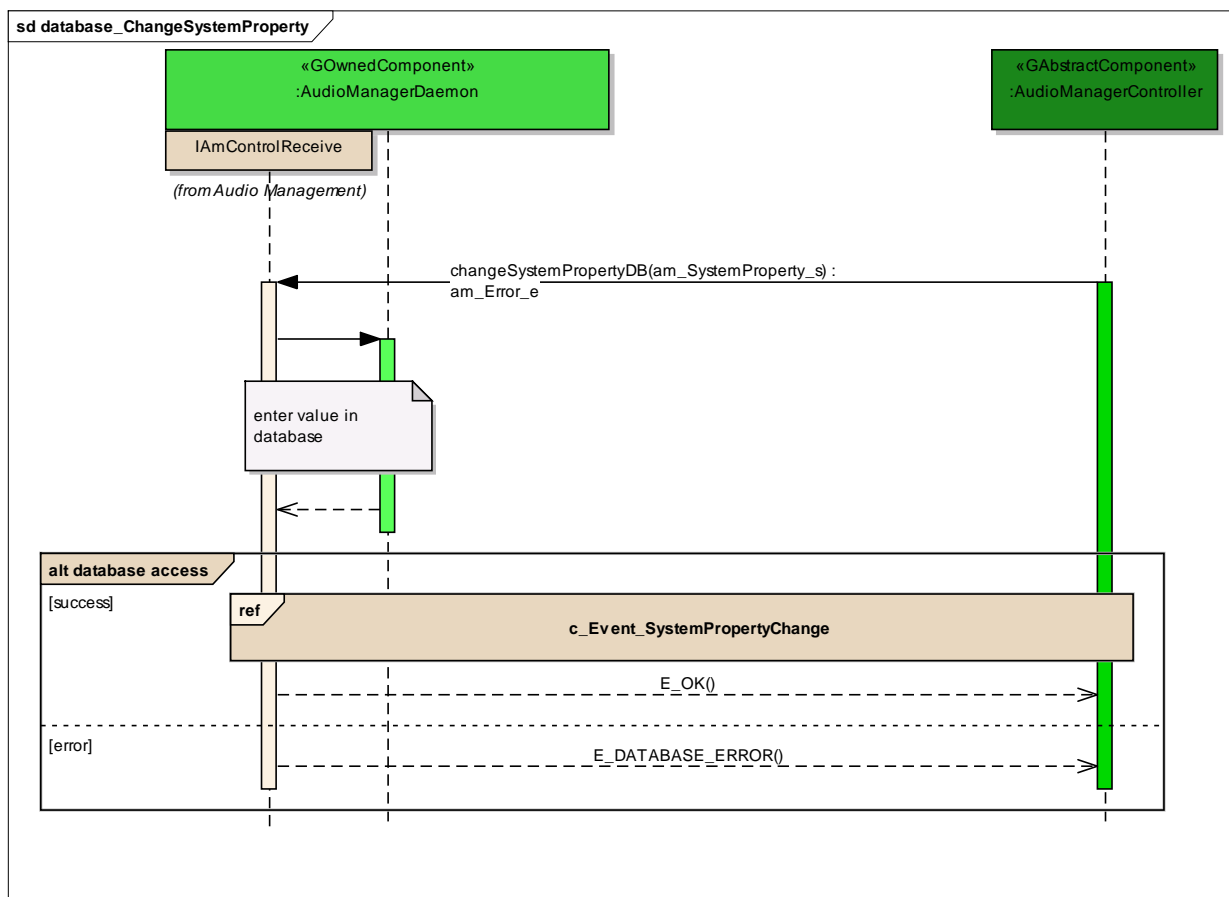
Figure: 51

c_Event_SourceSoundPropertyChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeSystemProperty *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{B4ED3E39-DFD2-40e8-AF32-A3277850725B}

c_Event_SystemPropertyChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeSystemProperty *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*

GUID:	{E604E5C7-0421-4934-8E66-936047527999}

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeSystemProperty *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{BE504407-8A86-4de6-9703-2B527E6A1D11}

database_ChangeMainVolme

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{1EB70720-C947-4b44-9D16-EA3D9259865B}

database_ChangeMainVolme - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{61FB65C7-DF6B-4001-AD90-7F8543CDA922}
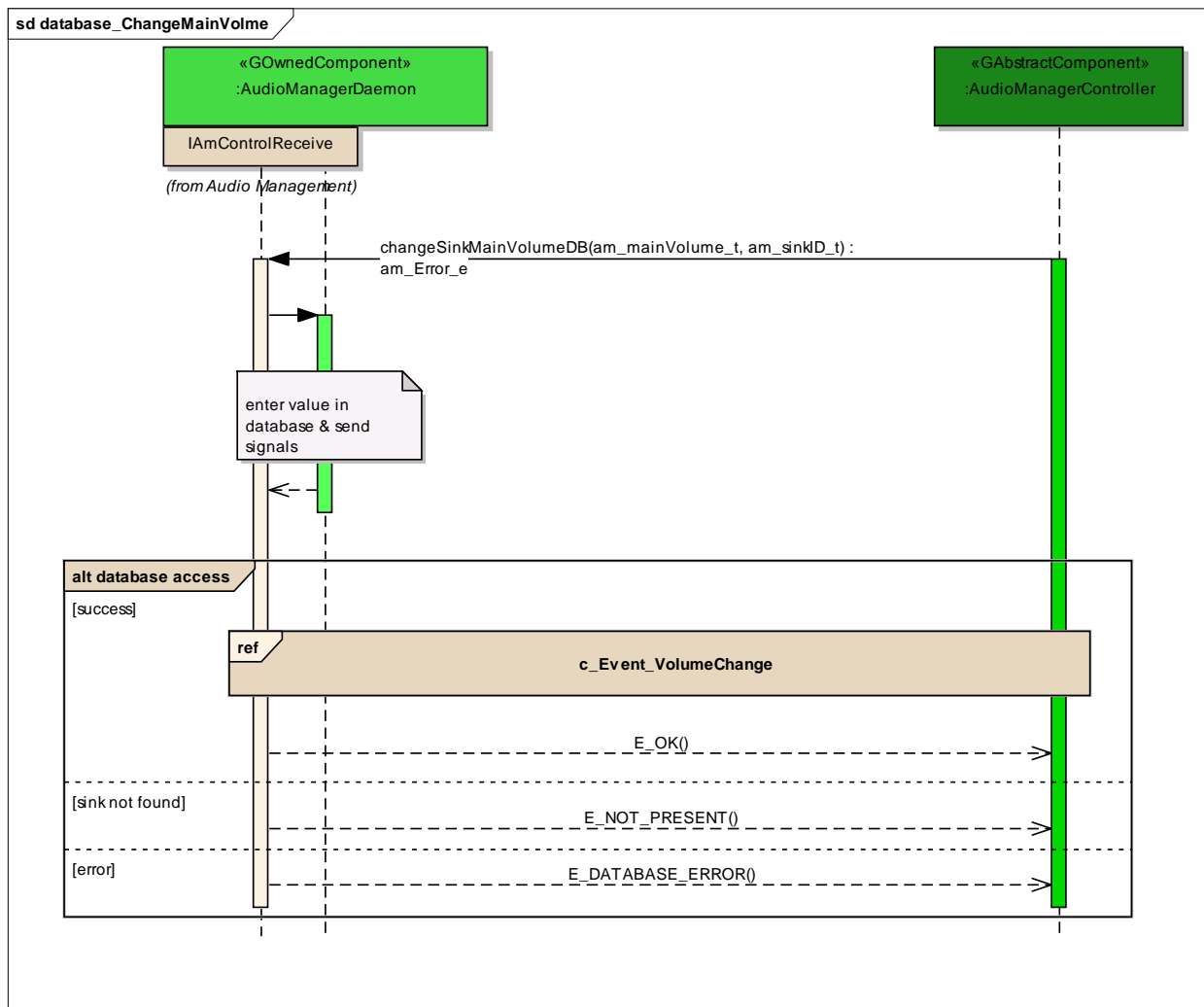
Figure: 52

c_Event_VolumeChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMainVolme *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{DD88E74D-5F28-41e0-B41C-62C394B12C95}

database access

Type:	**InteractionFragment**

Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeMainVolme *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{C8C44520-F53E-443e-A9CA-E18A3BF19D18}

database_ChangeDomainState

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{1AC6D9DF-48DB-4ad6-A35E-AEC3AD188CF9}

database_ChangeDomainState - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{38972680-16F8-4ebf-AF67-65619A564F58}
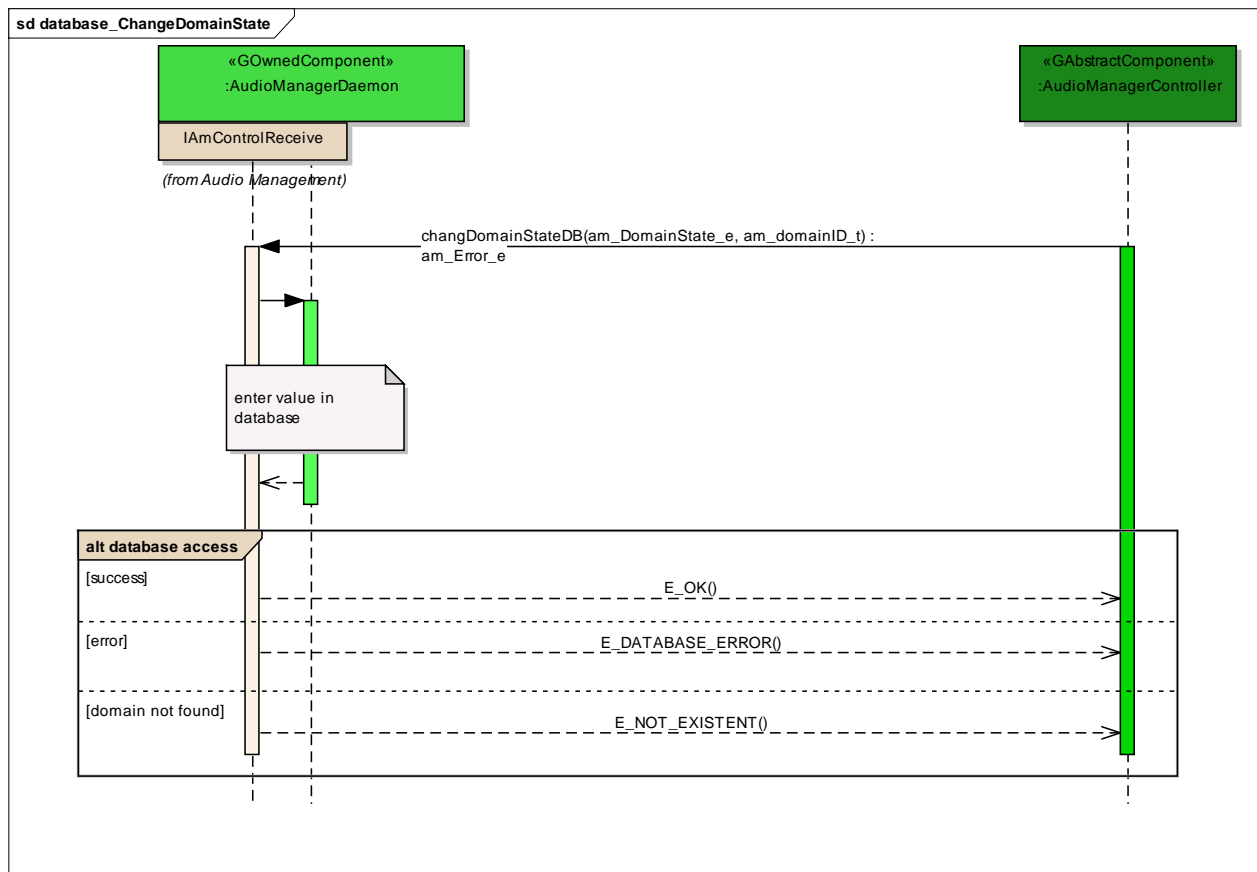
Figure: 53

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeDomainState *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{92D95688-582C-4ee6-B640-2CCDA10B1796}

database_ChangeSinkAvailability

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*

database_ChangeSinkAvailability - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{9825589D-99C6-4944-B3FC-A5A5D8FA472B}
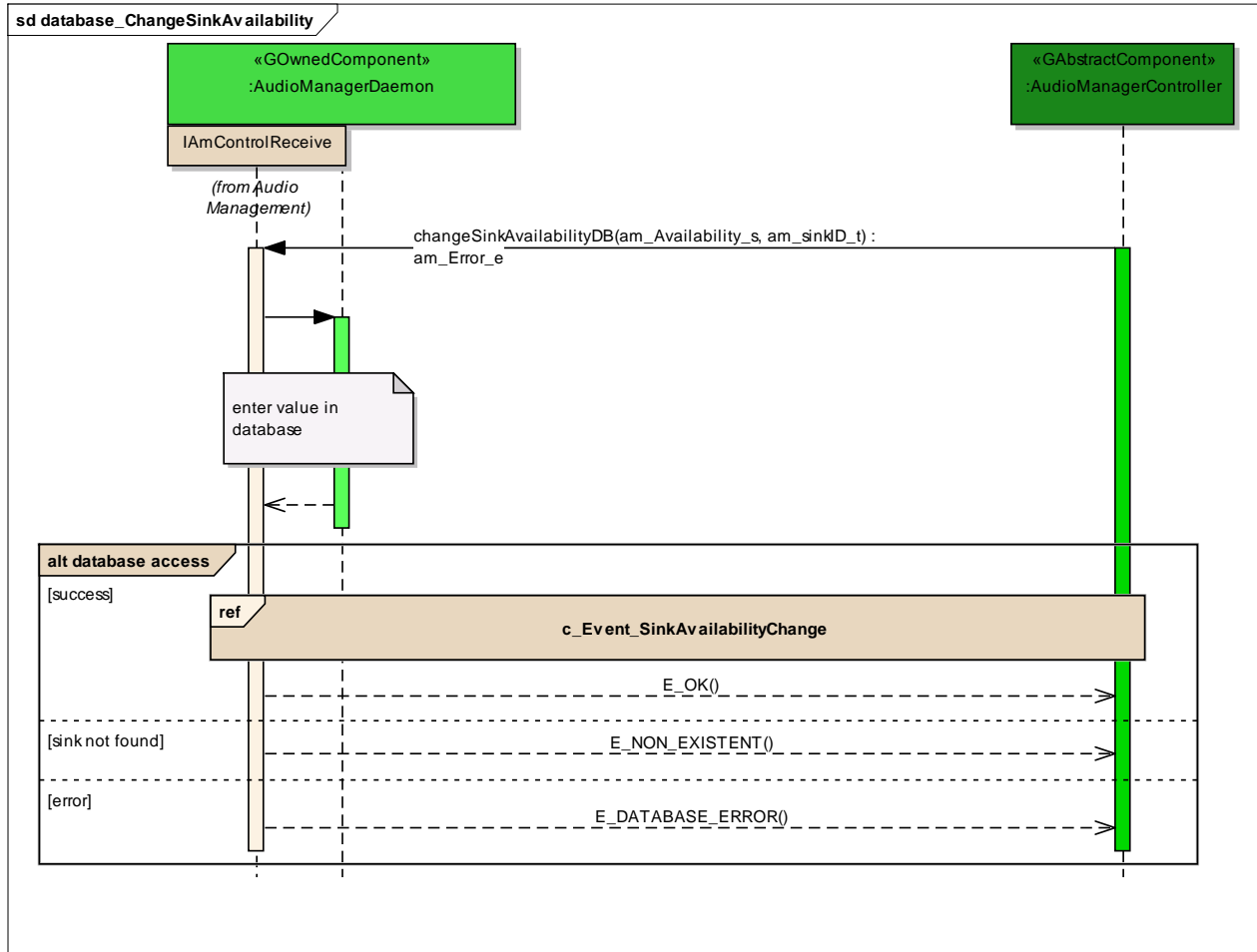
Figure: 54

c_Event_SinkAvailabilityChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeSinkAvailability *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{FF97EF65-6785-407c-8A6D-7B9DCFEC60CA}

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeSinkAvailability *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{BEC05E9E-24E8-4067-BF7D-E4D10D25FB01}

database_ChangeSourceAvailability

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{30D28AA5-E8DC-47e4-AC5D-4CD4EBECAB48}

database_ChangeSourceAvailability - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{06E6805C-E07C-4900-B559-4636D8C1CFA7}
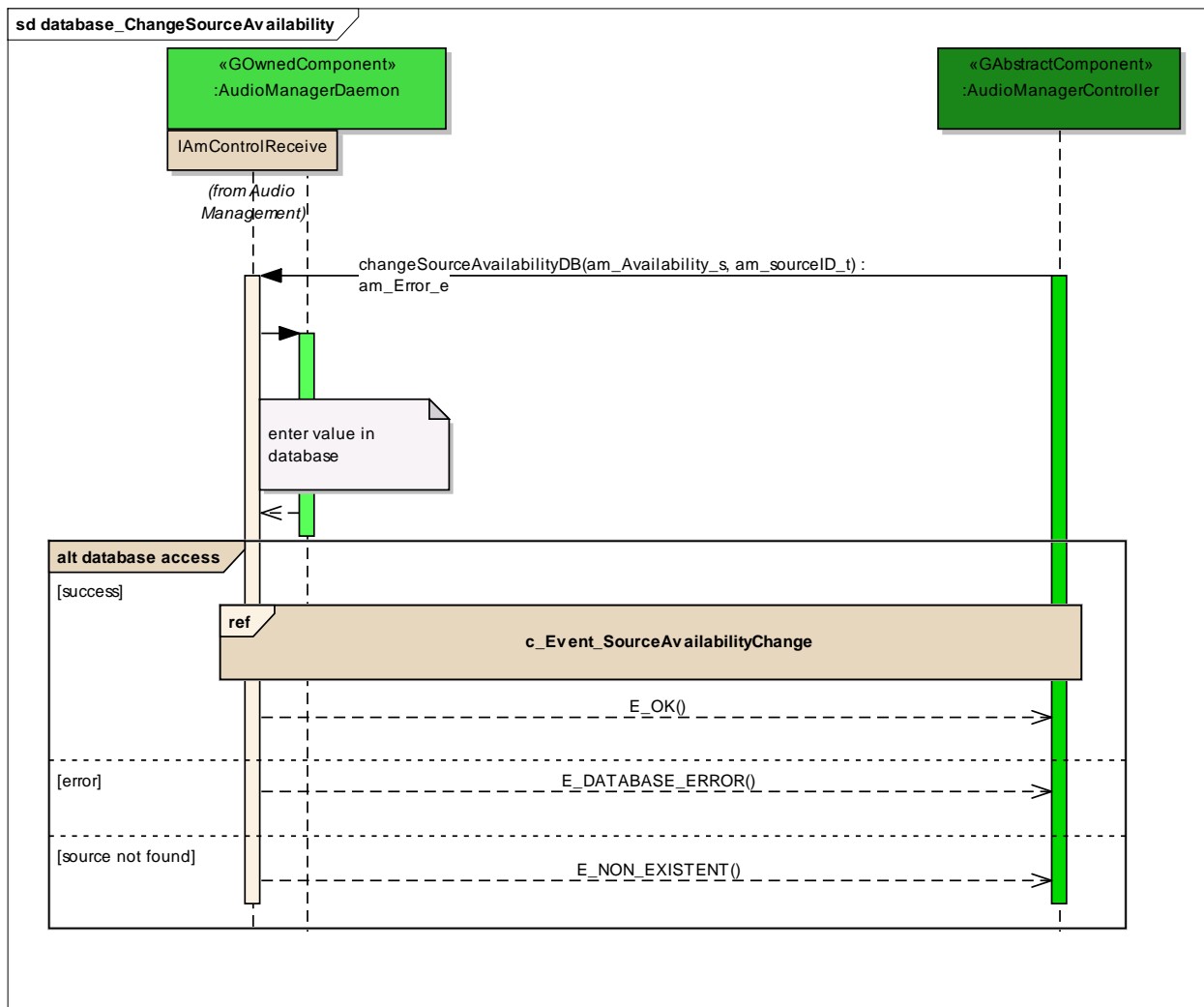
Figure: 55

c_Event_SourceAvailabilityChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeSourceAvailability *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{51F84917-57AA-4075-990C-897CB7B94FB6}

database access

Type:	**InteractionFragment**

Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_ChangeSourceAvailability *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{50AC8655-4642-4ee8-BC7E-A0D333B3A345}

database_EnterMainConnection

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 07.10.2011. Last modified on 10.10.2011*
GUID:	{C95D2EB5-BB35-4f94-814C-30AE1F4D0998}

database_EnterMainConnection - *(Interaction diagram)*

Created By:	christian *on* 07.10.2011
Last Modified:	14.02.2012
Version:	1.0. *Locked:* False
GUID:	{5DE0B5E5-25C2-4c42-9E63-CBA77DF6E014}
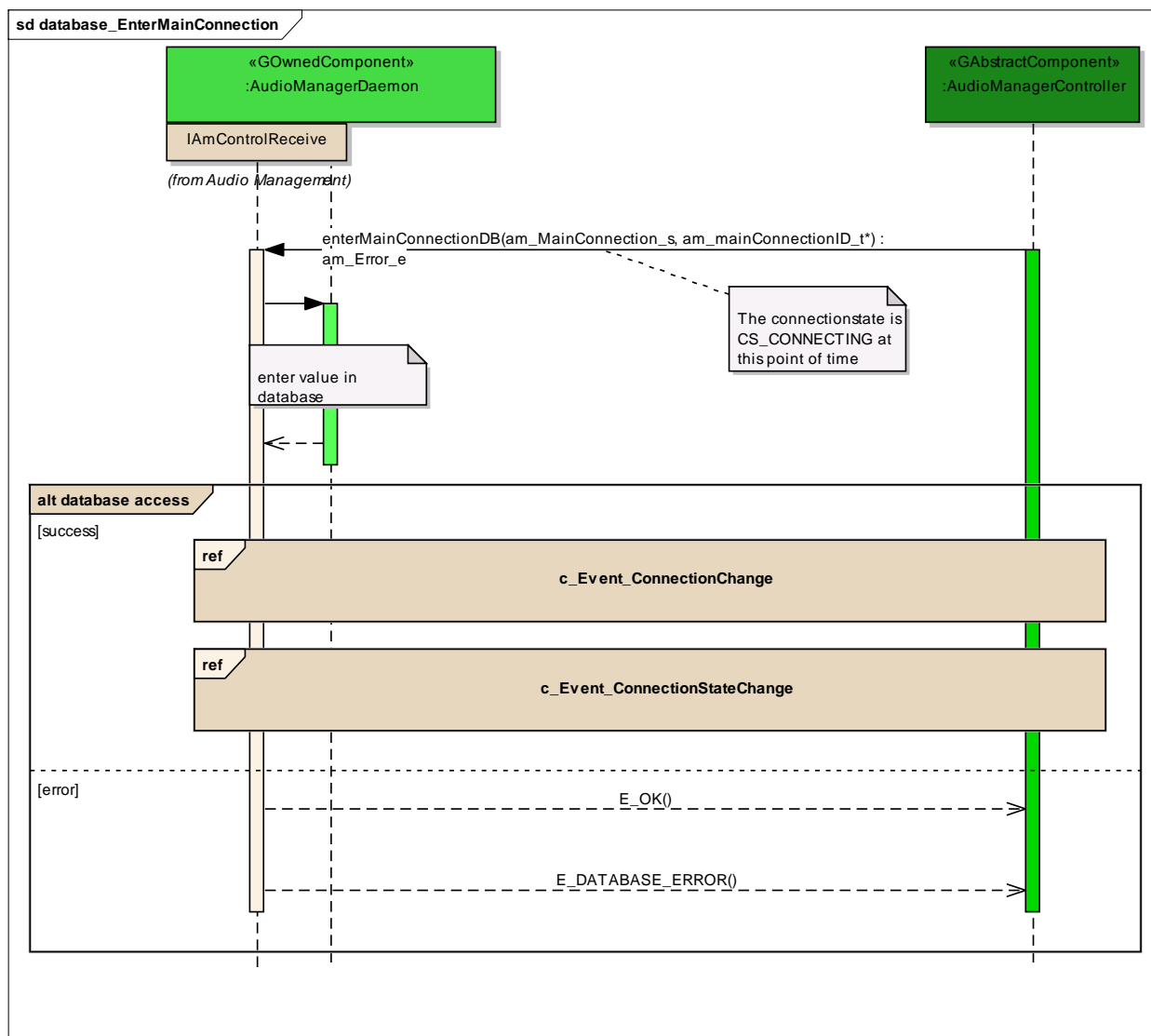
Figure: 56

c_Event_ConnectionChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterMainConnection *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{9E6B39E6-511D-4a46-9CC4-FB562797D9A8}

c_Event_ConnectionStateChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterMainConnection *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{00F8EEAF-173F-40c7-A3B6-08EB1B49B5B4}

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterMainConnection *Keywords:*
Detail:	*Created on 07.10.2011. Last modified on 07.10.2011.*
GUID:	{54414654-70E7-41f3-9880-95C9C495B14E}

database_EnterSink

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{CED74265-48B5-451e-A76E-56CCD5B4997D}

database_EnterSink - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	08.12.2011
Version:	1.0. *Locked:* False
GUID:	{20B73B85-B430-4744-87C1-A0D19B44C7E3}
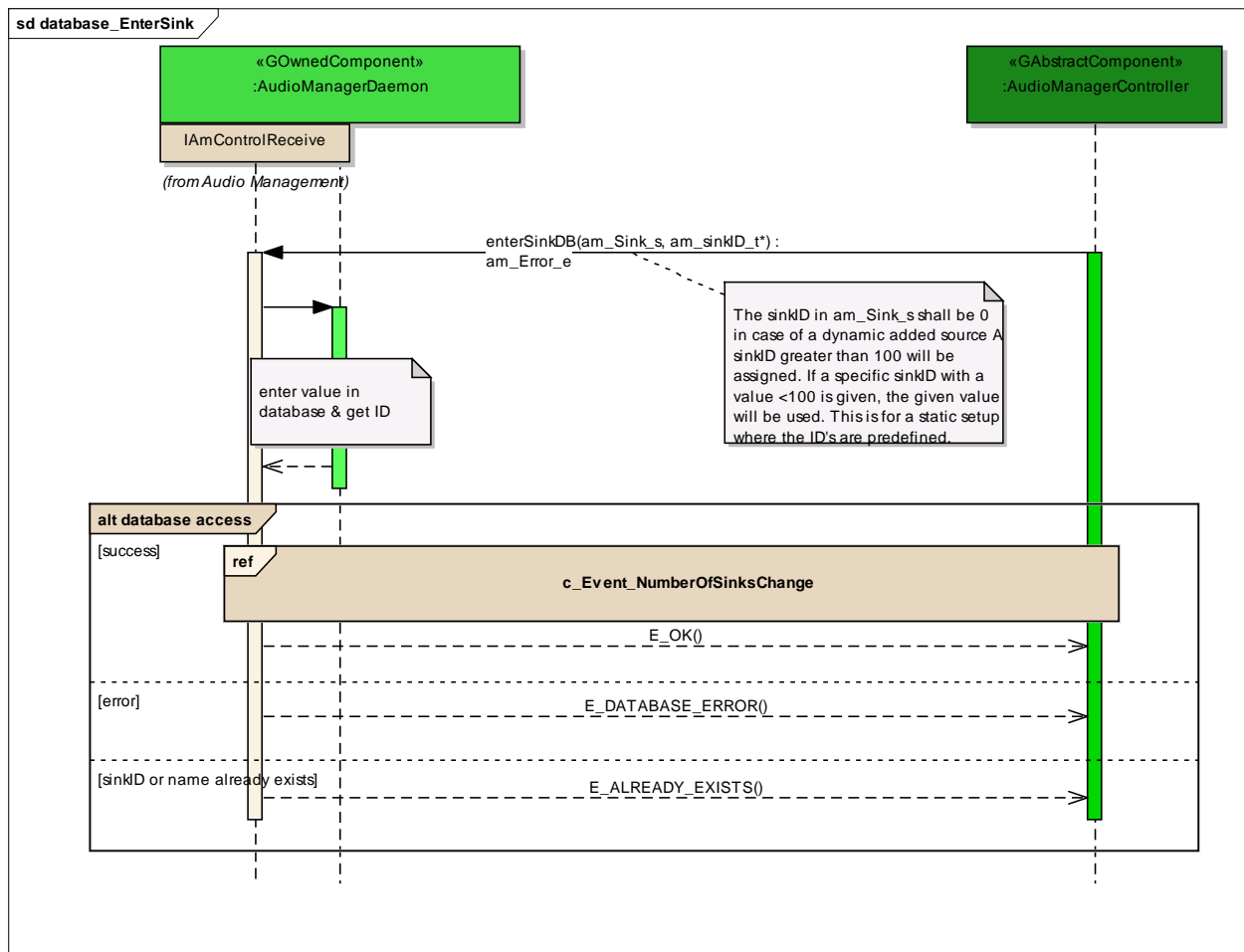
Figure: 57

c_Event_NumberOfSinksChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterSink *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{514B1DC5-ADC2-42ad-A55D-1B4C828FC21B}

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterSink *Keywords:*

database_EnterSource

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{6DA55940-5C85-4ef2-AEE0-27D041E1EFD1}

database_EnterSource - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{2F3A7934-591B-40d9-B8C9-E4652C1EF491}
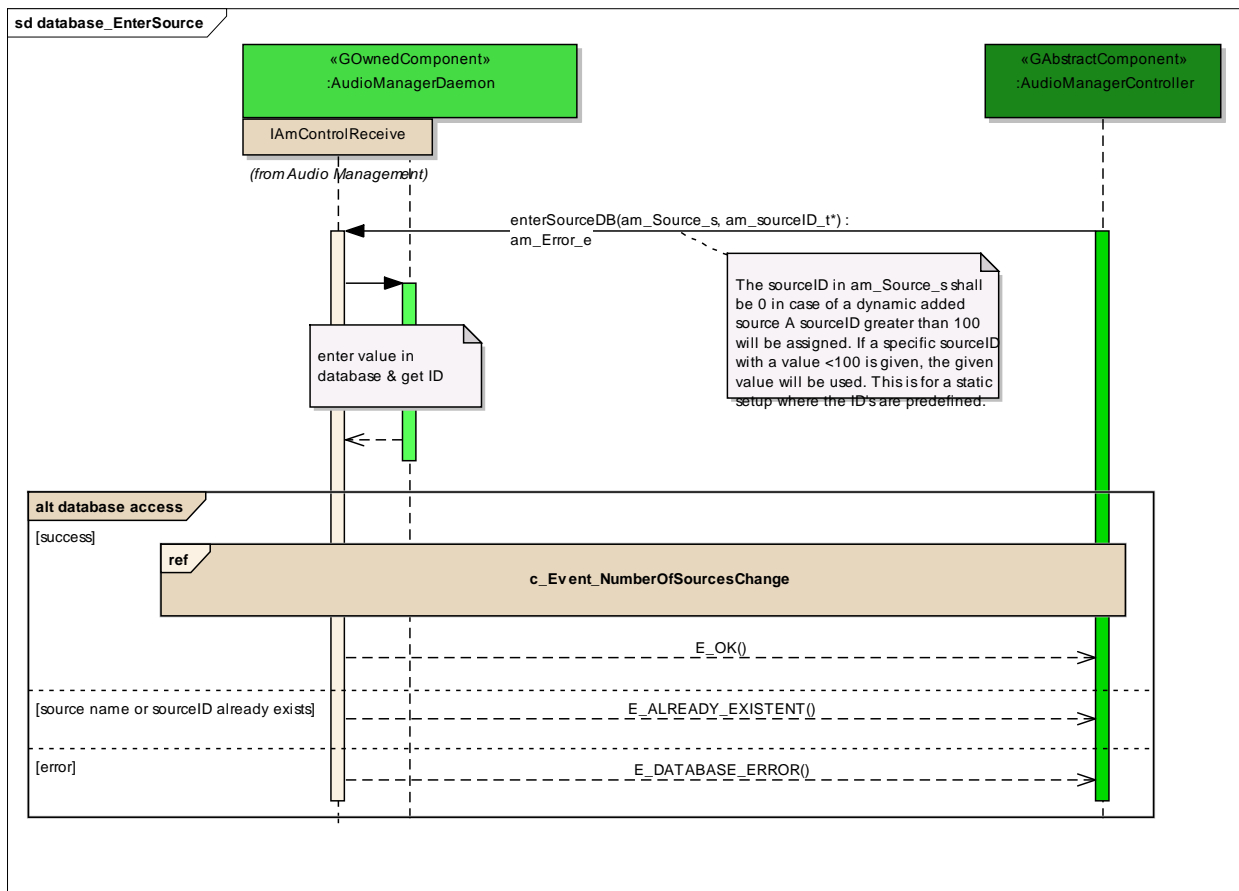
Figure: 58

c_Event_NumberOfSourcesChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterSource *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{F9DF8386-7F97-4281-8730-2D3ED52253F6}

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterSource *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{8B674DF9-5E4C-4404-94C0-37B1BA8FB1D9}

database_EnterDomain

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{DCEEC7D3-9FE4-4a12-9EB7-DDE68DCB1480}

database_EnterDomain - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{670674CD-AB3B-4447-BADE-4A968889AD8C}
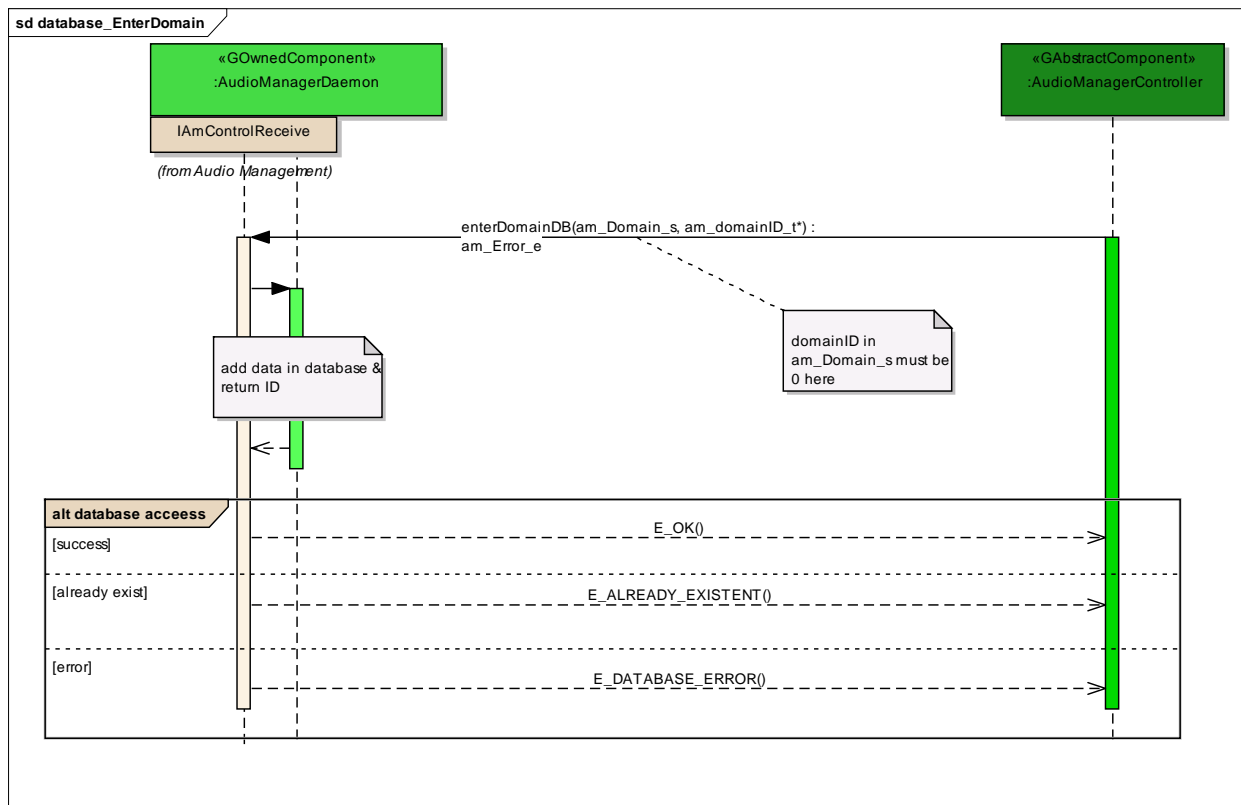
Figure: 59

database acceess

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterDomain *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{AE7CE3B9-D5FA-404e-B827-5D63F8FA061B}

database_EnterGateway

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{4DEDCF61-8B2D-4689-B686-A247AB779716}

database_EnterGateway - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{9F3811DD-E6DC-4b82-A978-1539C254EE82}
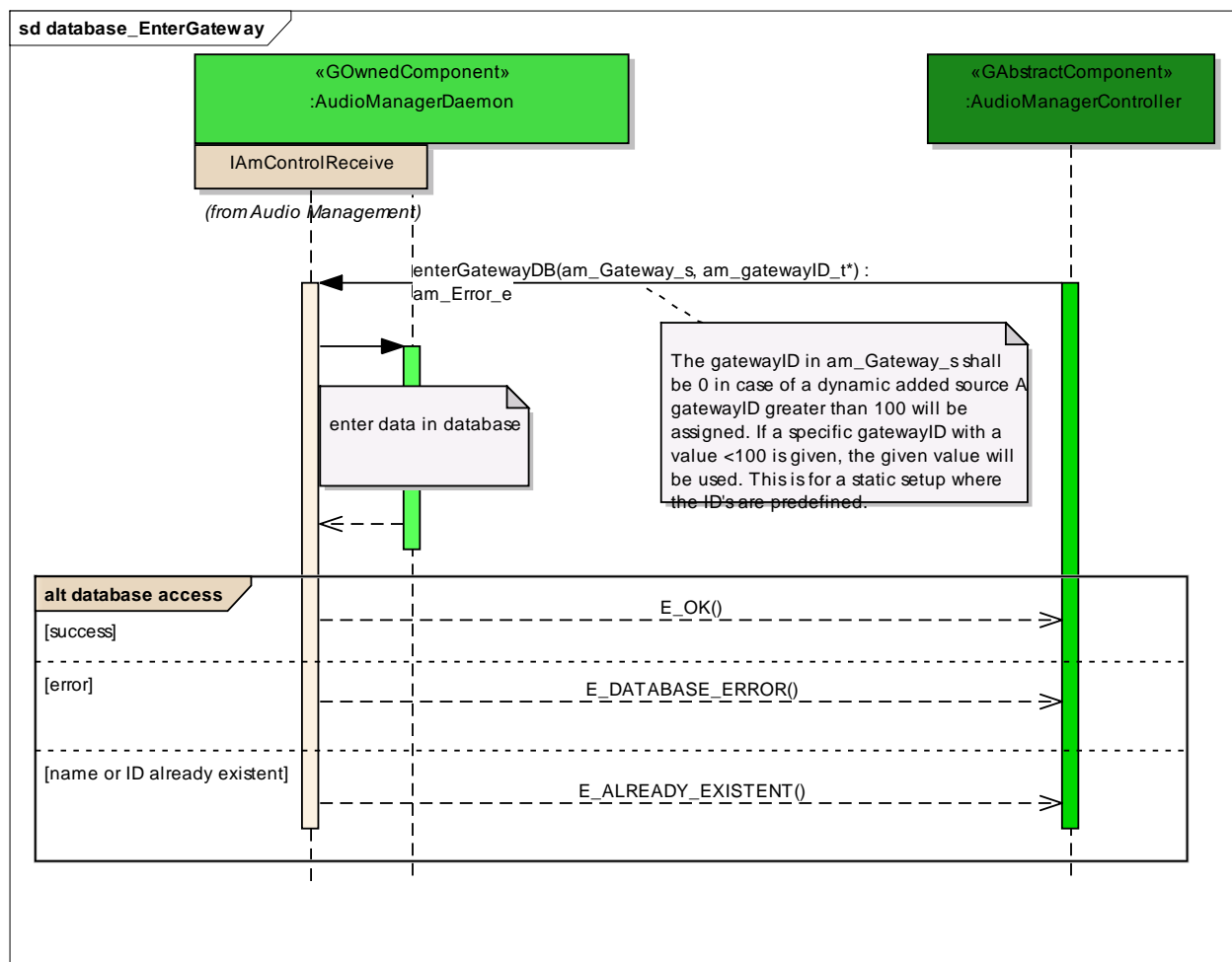
Figure: 60

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterGateway *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{89FF1E99-DCF8-461b-8574-4D24171F4B19}

database_EnterCrossfader

Type:	**Package**

Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011*
GUID:	{2987F3B7-6CB8-4490-B0DC-DAB94C96759F}

database_EnterCrossfader - *(Interaction diagram)*

Created By:	christian *on* 12.10.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{A55E0FC6-25F0-46fb-AE1A-6B7990E5021B}
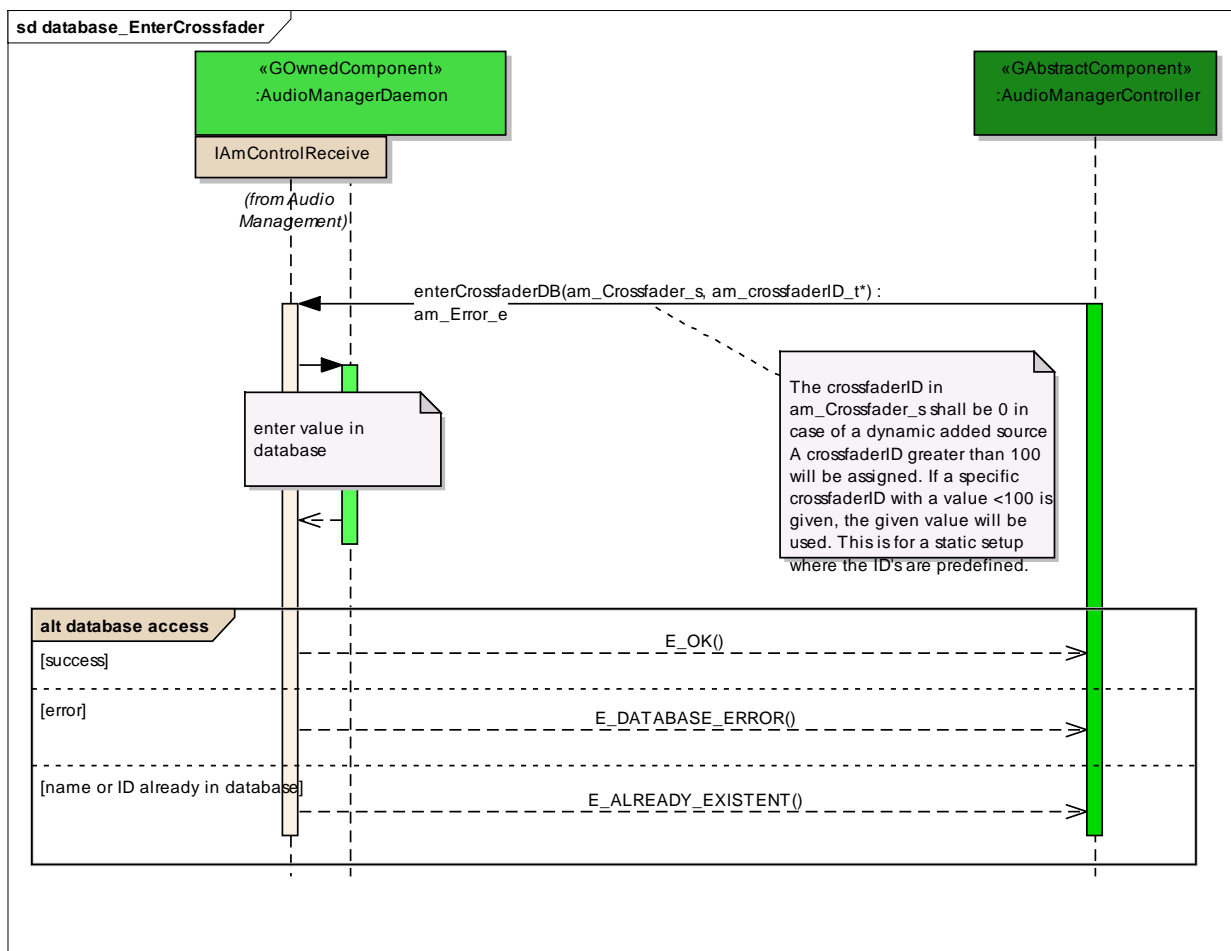
Figure: 61

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterCrossfader *Keywords:*

database_EnterSystemPropertiesList

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 07.12.2011. Last modified on 07.12.2011*
GUID:	{CA09420D-CD56-4fa5-9BB4-C6EE05E81A46}

database_EnterSystemPropertiesList - *(Interaction diagram)*

Created By:	christian *on* 07.12.2011
Last Modified:	07.12.2011
Version:	1.0. *Locked:* False
GUID:	{21C4DC87-4C6B-49fb-8834-8A6BDB8E49AD}
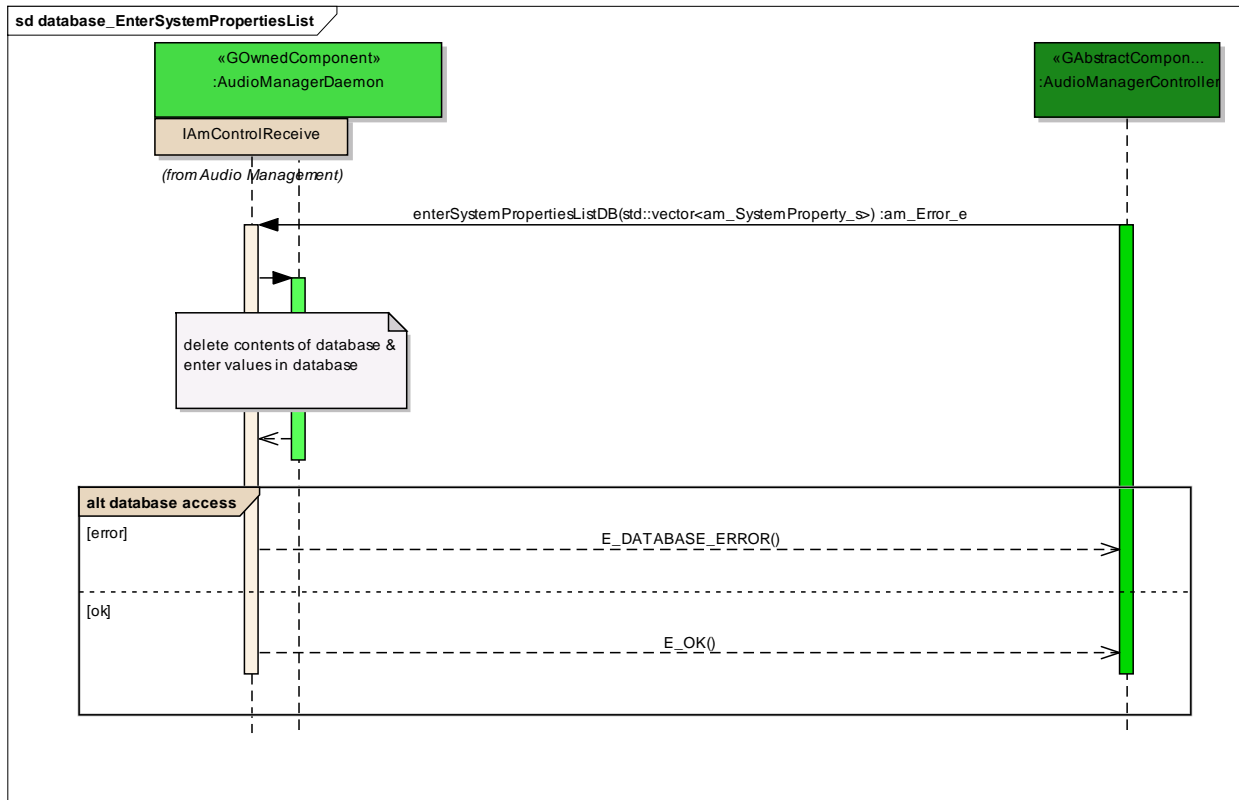
Figure: 62

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterSystemPropertiesList *Keywords:*
Detail:	*Created on 07.12.2011. Last modified on 07.12.2011.*
GUID:	{C7219531-214C-4507-9A0F-12B551B744E0}

database_EnterSinkClass

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 14.02.2012. Last modified on 14.02.2012*
GUID:	{46297306-8663-4f43-928A-3403D125343B}

database_EnterSinkClass - *(Interaction diagram)*

Created By:	christian *on* 08.12.2011
Last Modified:	14.02.2012
Version:	1.0. *Locked:* False
GUID:	{B1BA21EC-3873-4b8a-9E50-9EB61FC32BDB}
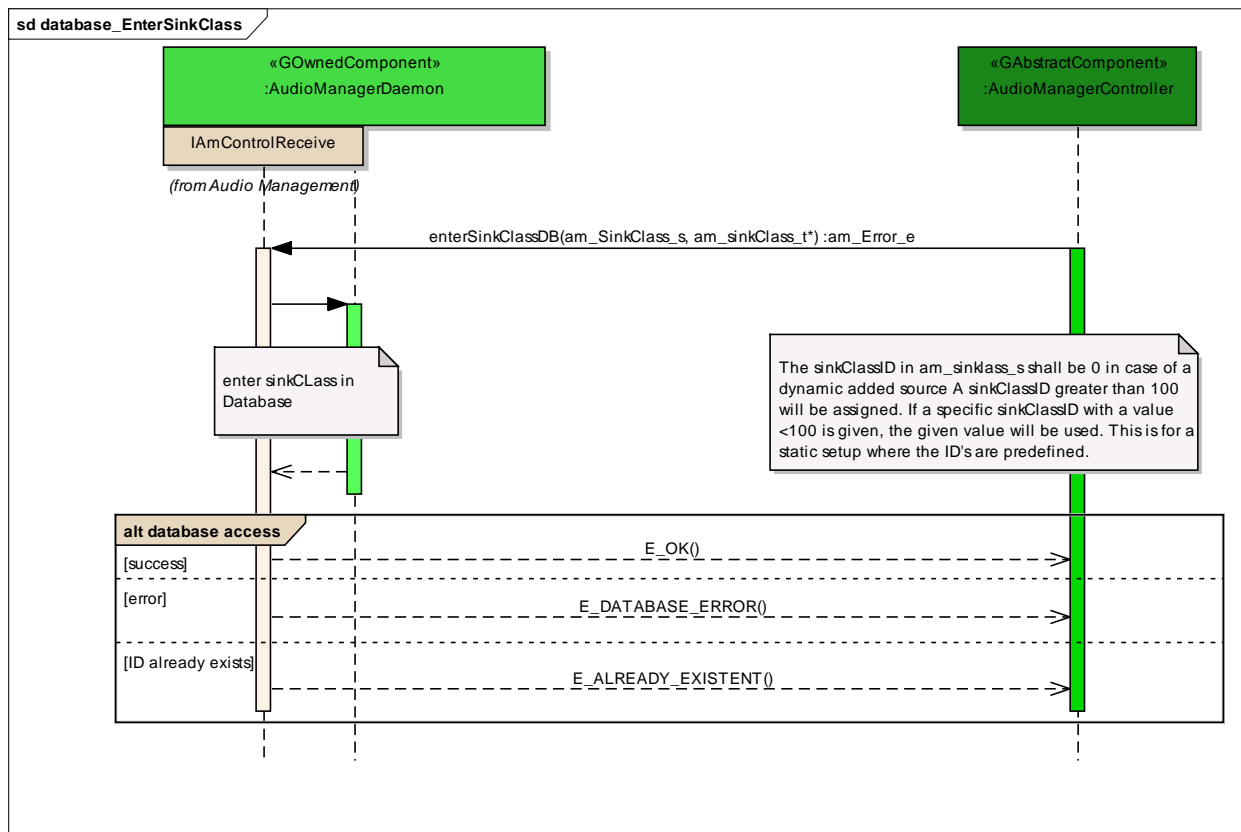
Figure: 63

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterSinkClass *Keywords:*
Detail:	*Created on 08.12.2011. Last modified on 08.12.2011.*
GUID:	{9C2881D8-A4E7-458f-BD1E-A880CBA0098E}

database_EnterSourceClass

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 08.12.2011. Last modified on 08.12.2011*
GUID:	{BC964E32-CCEA-43c0-AA75-DBA2BDD8EAC2}

database_EnterSourceClass - *(Interaction diagram)*

Created By:	christian *on* 08.12.2011
Last Modified:	14.02.2012
Version:	1.0. *Locked:* False
GUID:	{BF3583A3-1695-442c-9563-7C9128F0D8B7}
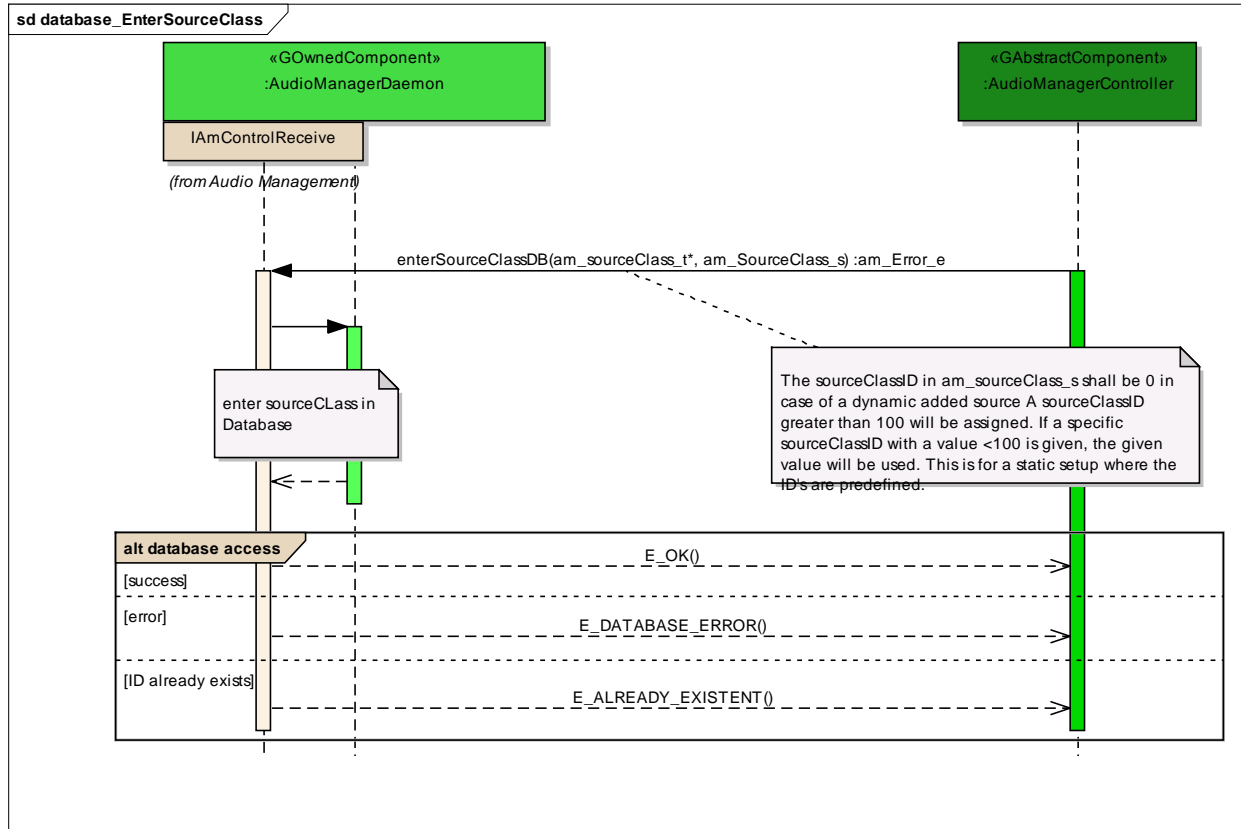
Figure: 64

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_EnterSourceClass *Keywords:*
Detail:	Created on 08.12.2011. Last modified on 08.12.2011.
GUID:	{1FC09DB0-ACE1-4c74-9F23-5DDE2CA2DC28}

database_RemoveMainConnection

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{B6E98F69-2493-4dc4-8850-AE5B2CA9F94E}

database_RemoveMainConnection - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{41328F26-9781-45eb-AB91-4FC251FA4380}
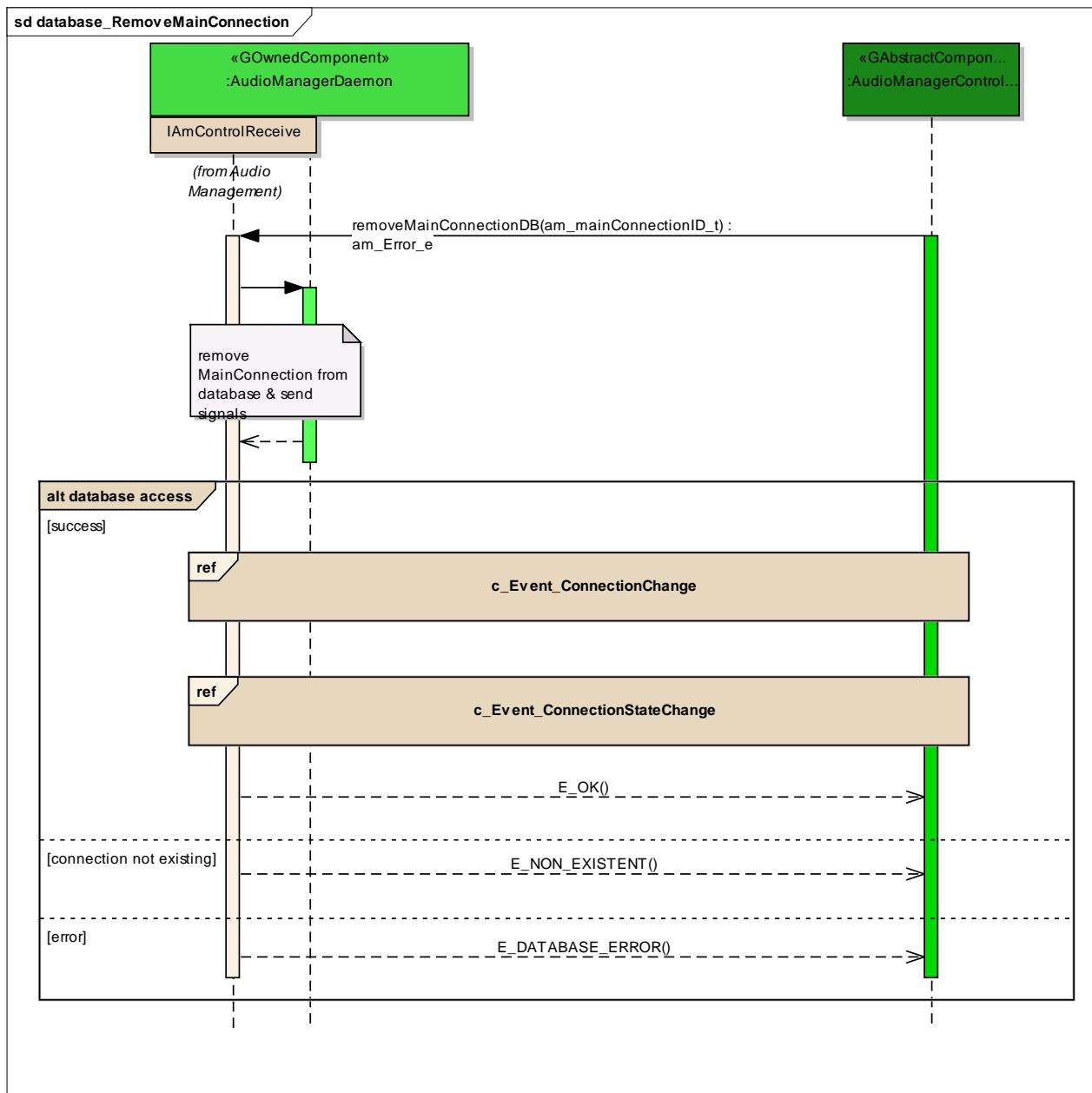
Figure: 65

c_Event_ConnectionChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_RemoveMainConnection *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{E257C9C2-1C3C-4491-B907-96B75116DF15}

c_Event_ConnectionStateChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_RemoveMainConnection *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{D8E73B4D-9B4A-4a87-AFD9-5D49BCF1DC43}

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_RemoveMainConnection *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{FCCEAB7D-FF84-4fd5-B266-BEB5B68D6BCA}

database_RemoveSink

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{4954311F-0DC3-495d-B858-725356469C24}

database_RemoveSink - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{CFE83F59-9143-4789-A524-989B608FAB75}
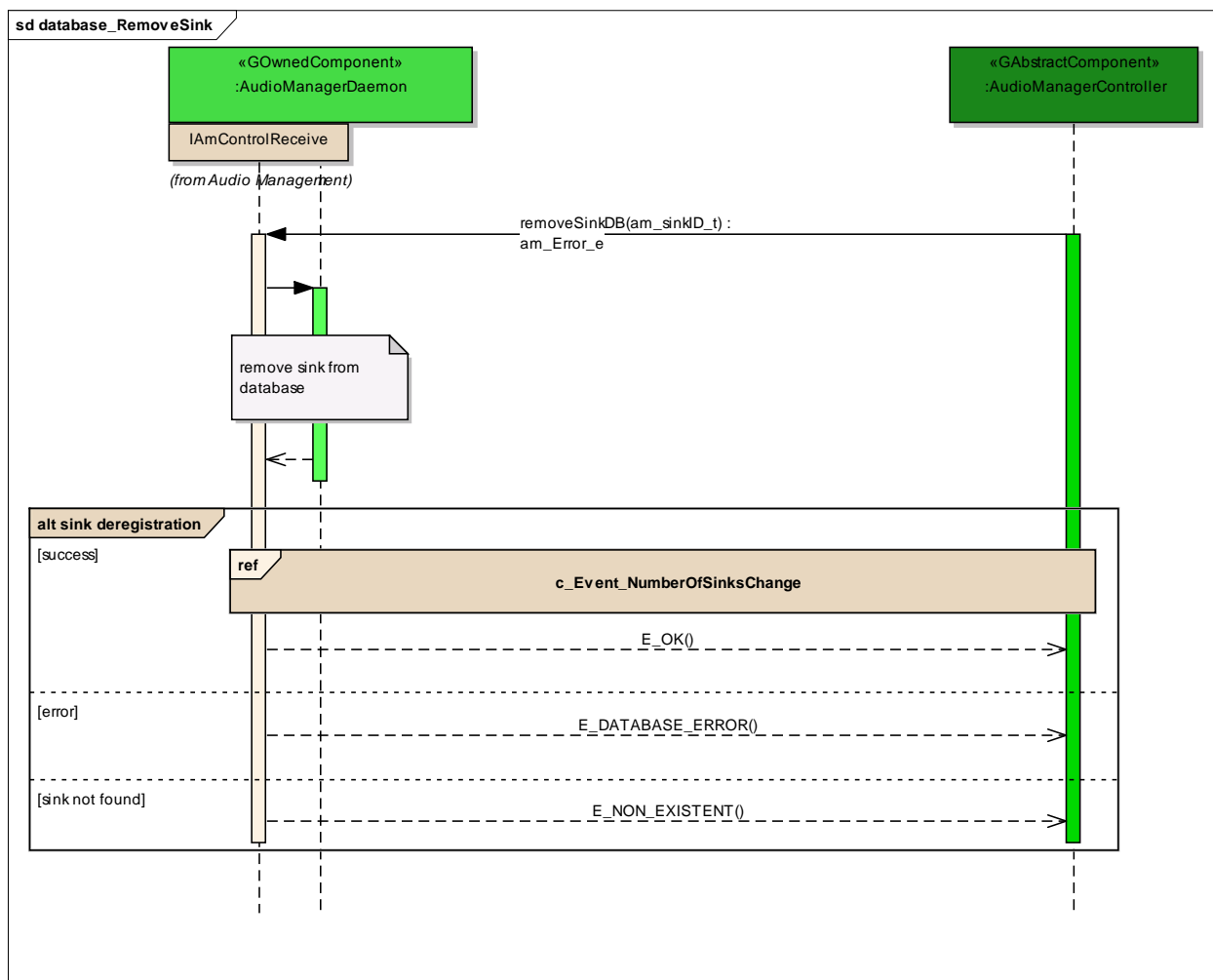
Figure: 66

c_Event_NumberOfSinksChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_RemoveSink *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{6950FD2D-7484-4c22-87FD-B382BC237636}

sink deregistration

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.

Package:	database_RemoveSink	*Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*	
GUID:	{C5FC93B7-C3A2-4906-81F0-F4E3C6867D86}	

database_RemoveSource

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{1BD4DFEA-D6DE-4640-BF17-F7776CA28AF5}

database_RemoveSource - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{1B38BFA1-83C4-465a-8514-4D36CEE04F22}
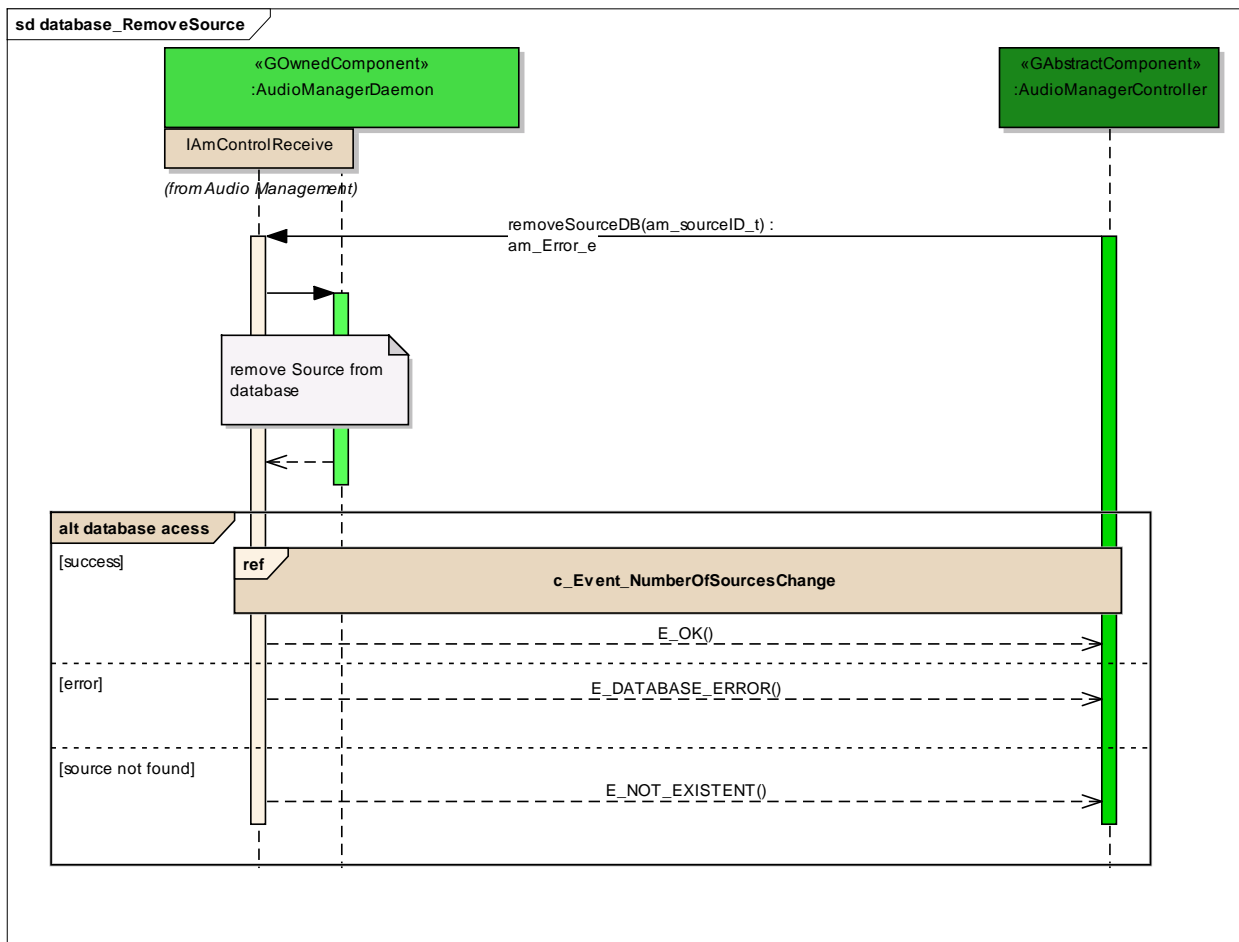
Figure: 67

c_Event_NumberOfSourcesChange

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_RemoveSource *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{DE1A798C-9020-4665-B4A2-9A915B4C2724}

database acess

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_RemoveSource *Keywords:*

Detail: *Created on 10.10.2011. Last modified on 10.10.2011.*
GUID: {300B5711-ED36-4aad-86A3-2AAA1BCE2649}

database_RemoveDomain

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011*
GUID:	{23455BA3-683C-42bd-AFB2-8DAD10205013}

database_RemoveDomain - *(Interaction diagram)*

Created By:	christian *on* 10.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{1D9B34B8-41A6-413e-A99E-A9C588A45310}
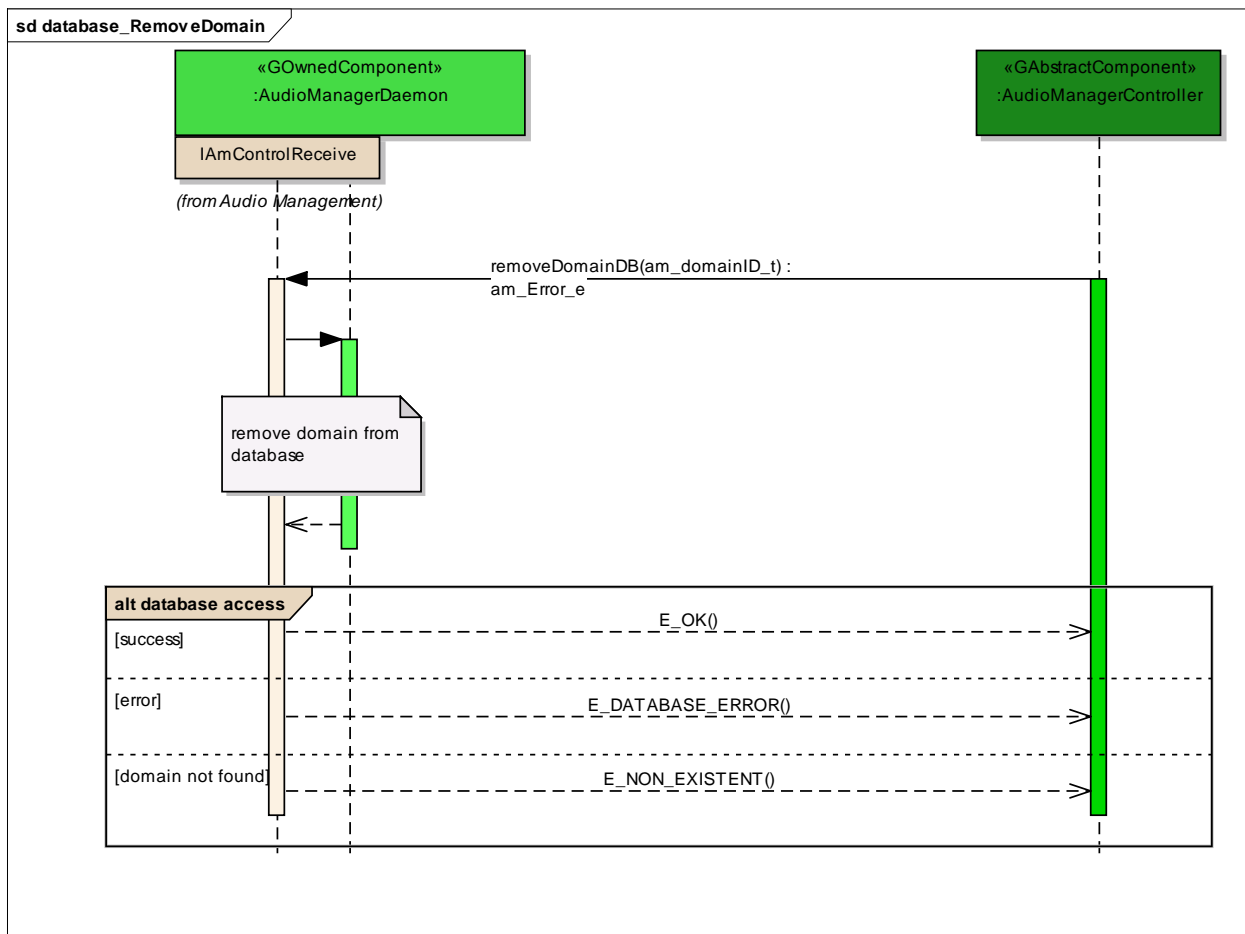
Figure: 68

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_RemoveDomain *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{06A0CA7F-86ED-496d-813A-4F7E964822CF}

database_RemoveGateway

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Database Access

database_RemoveGateway - *(Interaction diagram)*
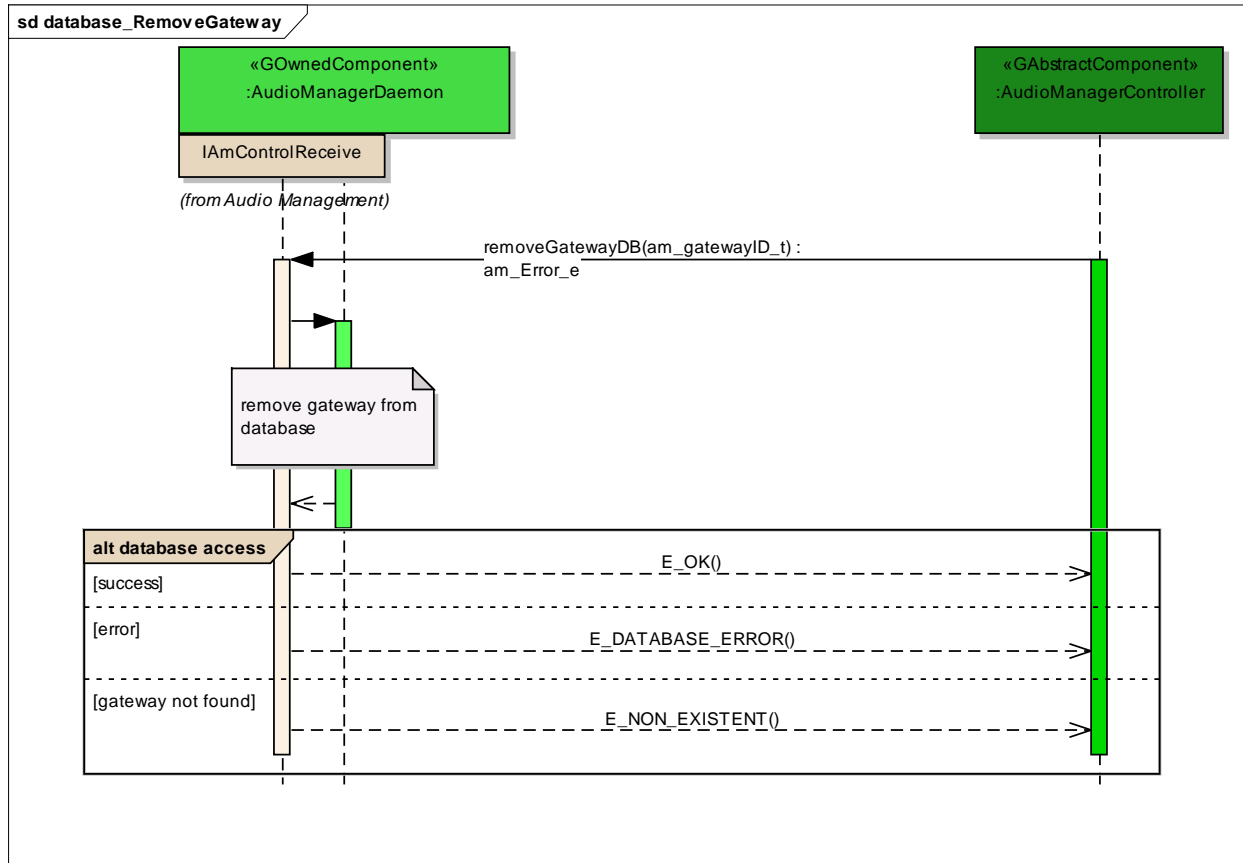
Figure: 69

database access

database_RemoveCrossfader

Type: **Package**
Status: Proposed. Version 1.0. Phase 1.0.
Package: Database Access
Detail: *Created on 12.10.2011. Last modified on 12.10.2011*
GUID: {228F2C3C-7FBF-4baa-89B1-33DAA17FF0D3}

database_RemoveCrossfader - *(Interaction diagram)*

Created By: christian *on* 12.10.2011
Last Modified: 08.12.2011
Version: 1.0. *Locked:* False
GUID: {7053B348-7F3B-4de8-9FFA-D80D15C2206F}
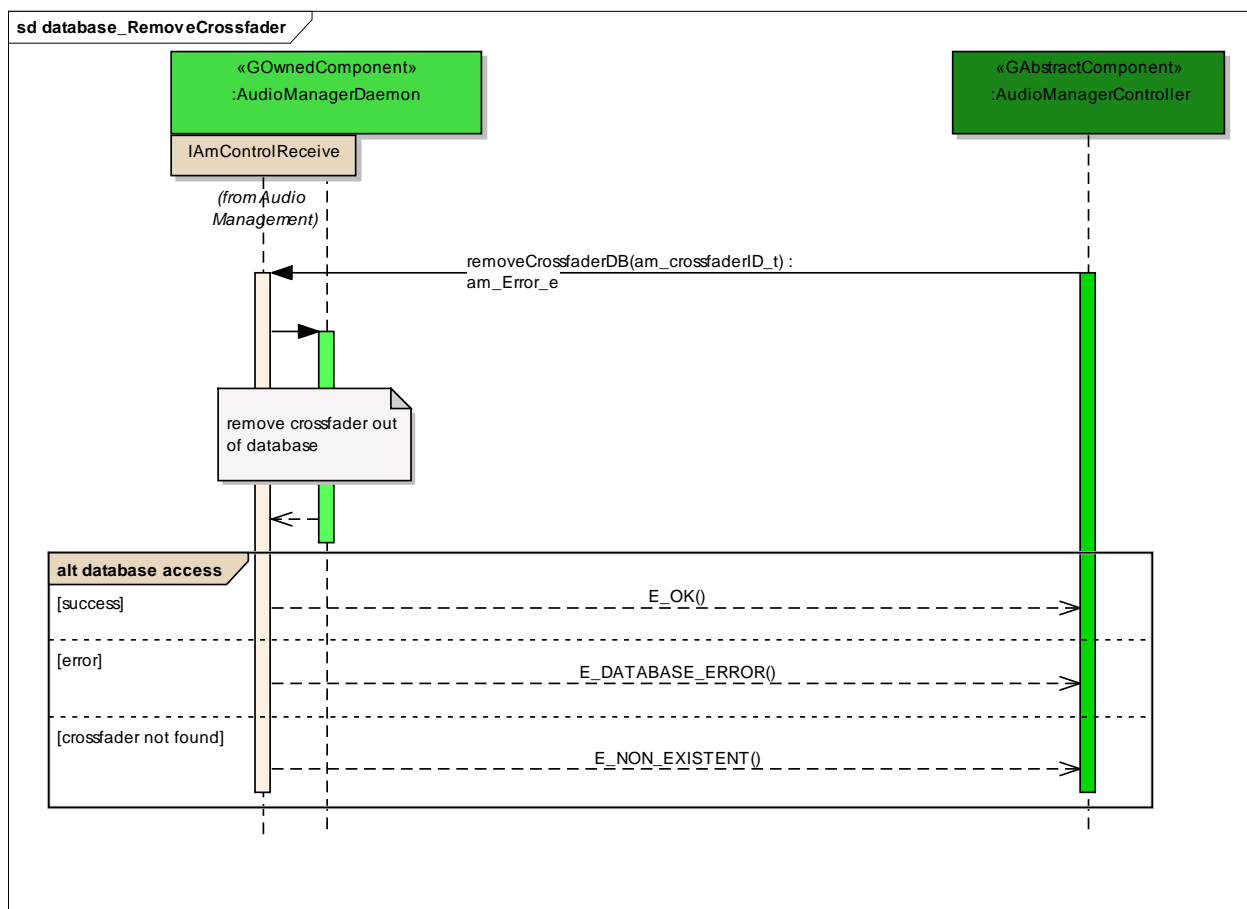
Figure: 70

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_RemoveCrossfader *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{E66D95D2-1B8C-4bdb-87A2-4EF1FE987A3F}

database_RemoveSinkClass

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 08.12.2011. Last modified on 08.12.2011*
GUID:	{DCF2309E-D2C4-4b10-9C1D-8DF89471D400}

database_RemoveSinkClass - *(Interaction diagram)*

Created By:	christian *on* 08.12.2011
Last Modified:	08.12.2011
Version:	1.0. *Locked:* False
GUID:	{DA928728-4C31-45a3-9745-77E4131C5C54}
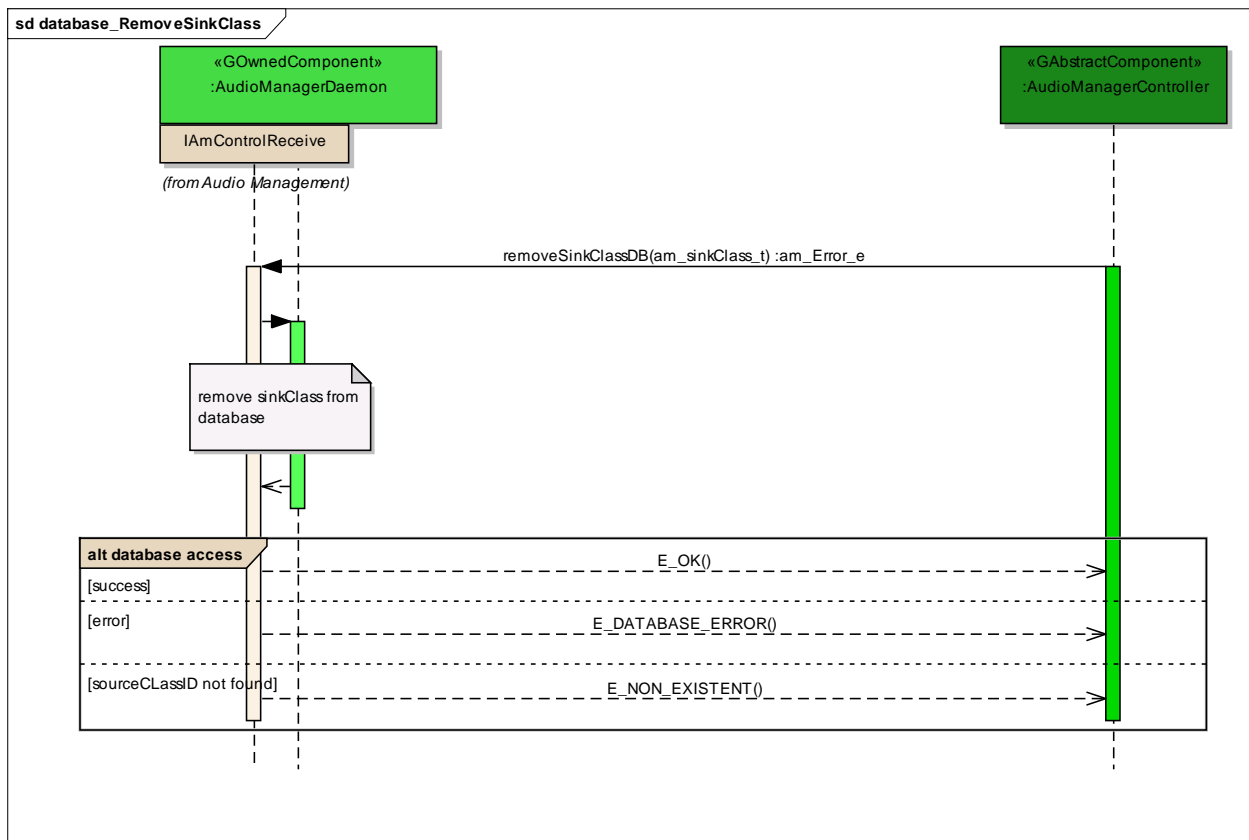
Figure: 71

database access

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_RemoveSinkClass *Keywords:*
Detail:	*Created on 08.12.2011. Last modified on 08.12.2011.*
GUID:	{06A5C533-9EA3-4ca5-9492-36721A6F49E5}

database_RemoveSourceClass

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 08.12.2011. Last modified on 08.12.2011*
GUID:	{E335023E-AD6A-48a3-B163-A4306E27DC42}

database_RemoveSourceClass - *(Interaction diagram)*

Created By: christian *on* 08.12.2011
Last Modified: 08.12.2011
Version: 1.0. *Locked:* False
GUID: {DB09A6E5-35EF-46a7-9029-FC901A459573}
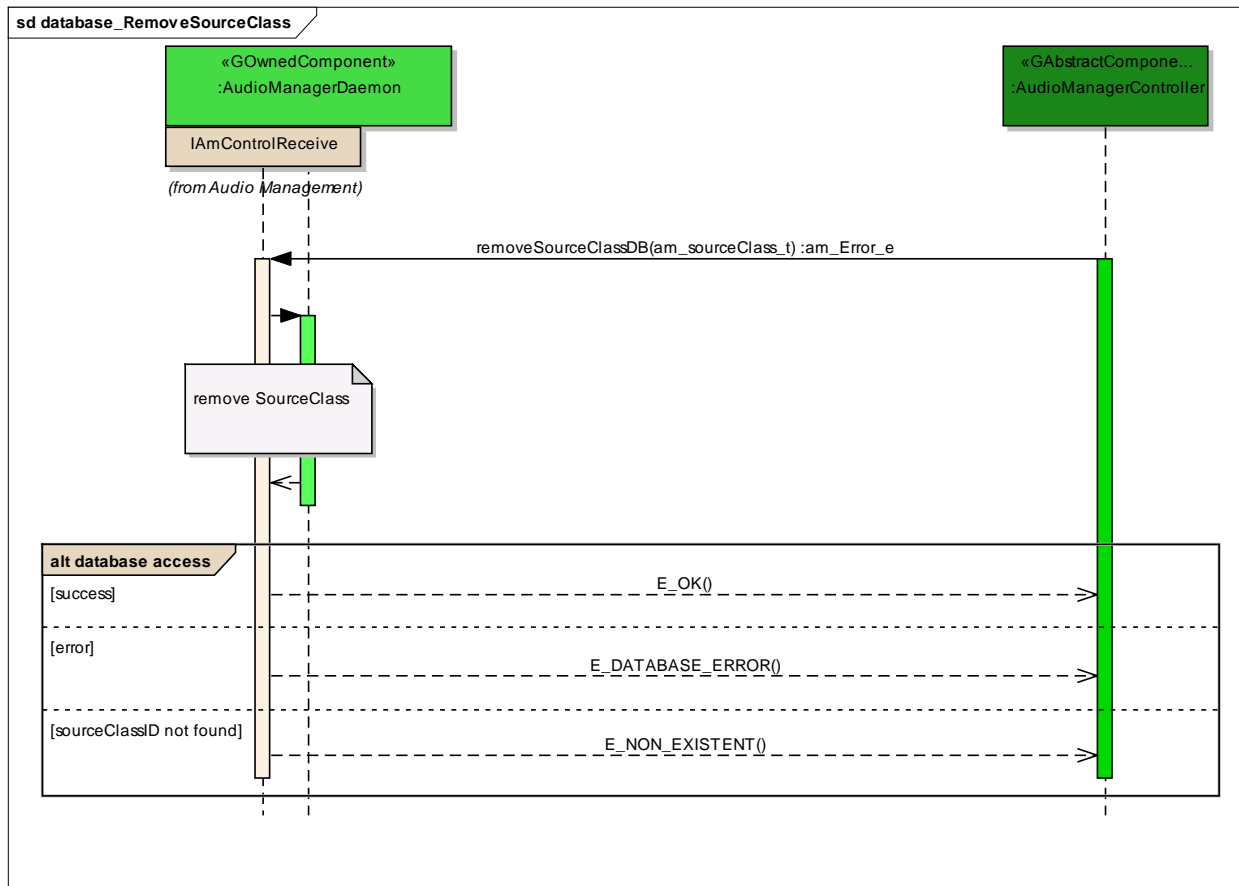
Figure: 72

database access

Type: **InteractionFragment**
Status: Proposed. Version 1.0. Phase 1.0.
Package: database_RemoveSourceClass   *Keywords:*
Detail: *Created on 08.12.2011. Last modified on 08.12.2011.*
GUID: {FF238EAF-5418-4762-9A10-87D539B7AB1F}

database_UpdateSources

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 28.06.2012. Last modified on 29.06.2012*
GUID:	{EFC1E2A0-055B-436f-9E33-65D099A96B56}

database_UpdateSources - *(Interaction diagram)*

Created By:	q232968 *on* 28.06.2012
Last Modified:	28.11.2012
Version:	1.0. *Locked:* False
GUID:	{9E108917-2A16-4756-AA75-682EF9EF7DC2}
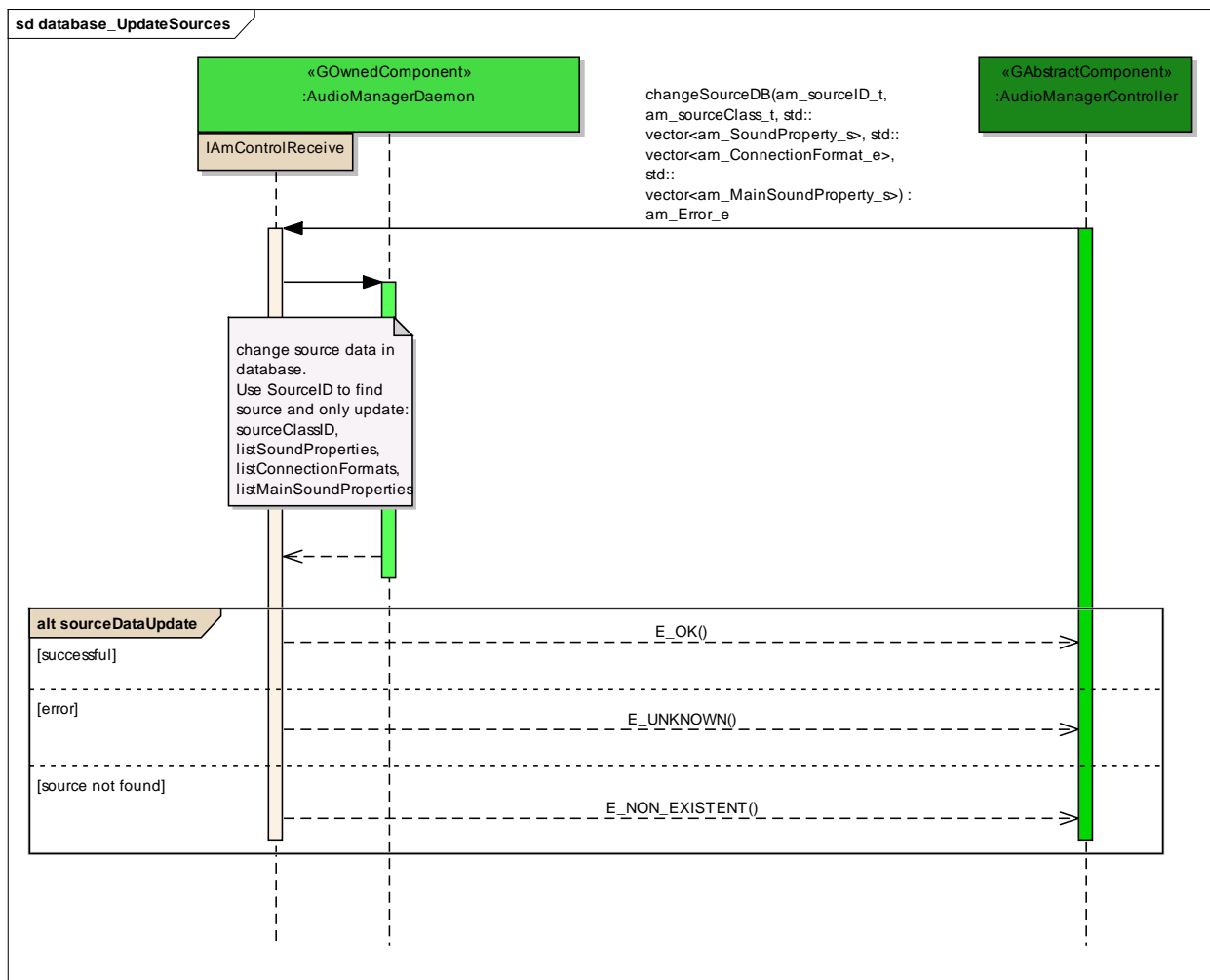
Figure: 73

sourceDataUpdate

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_UpdateSources *Keywords:*
Detail:	*Created on 28.06.2012. Last modified on 28.06.2012.*
GUID:	{30F30404-17D6-430b-963A-0EB9C39B5D0D}

database_UpdateSinks

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 29.06.2012. Last modified on 29.06.2012*
GUID:	{03BDDD22-1CB3-4a6b-94DE-B22C07399D18}

database_UpdateSinks - *(Interaction diagram)*

Created By:	q232968 *on* 28.06.2012
Last Modified:	29.06.2012
Version:	1.0. *Locked:* False
GUID:	{D2A39AA0-F723-4b23-B1DF-34044FFA8B52}
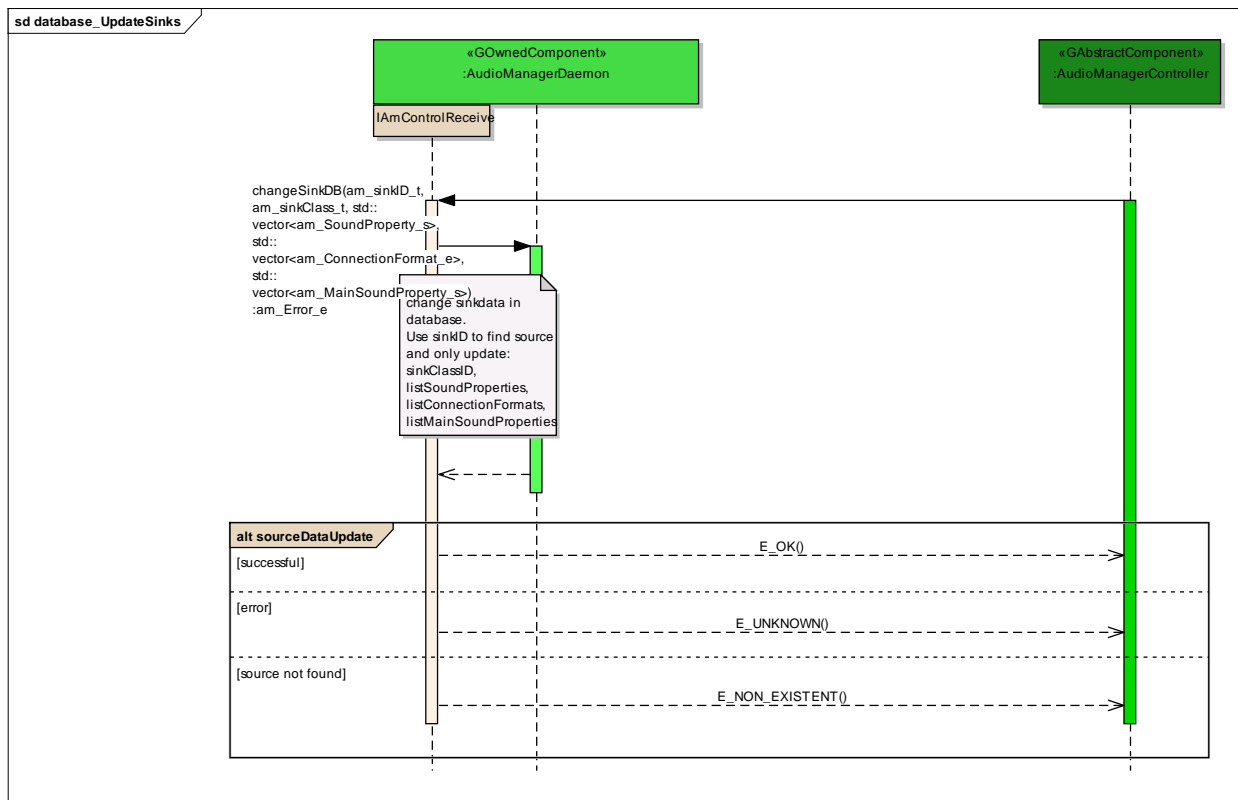
Figure: 74

sourceDataUpdate

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_UpdateSinks *Keywords:*
Detail:	*Created on 28.06.2012. Last modified on 28.06.2012.*
GUID:	{B91D6514-8C2B-4380-913B-E844979EA7AA}

database_UpdateGateways

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	Database Access
Detail:	*Created on 29.06.2012. Last modified on 29.06.2012*
GUID:	{C8134FA1-B765-4bef-8D7A-5475D3574ABB}

database_UpdateGateways - *(Interaction diagram)*

Created By:	q232968 *on* 28.06.2012
Last Modified:	29.06.2012
Version:	1.0. *Locked:* False
GUID:	{1C57B2CD-0E3F-49b5-BA94-ACBCAC952DCF}
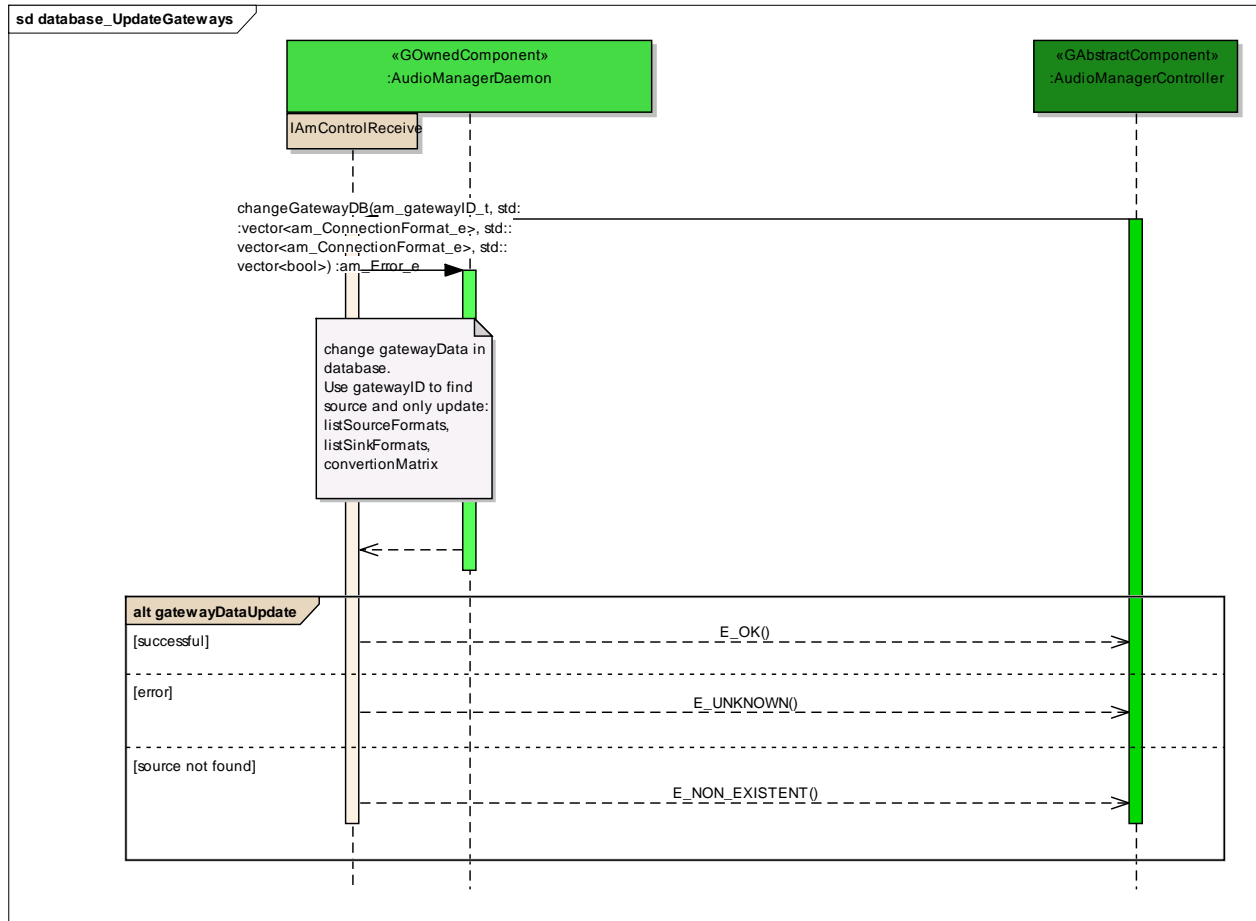
Figure: 75

gatewayDataUpdate

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	database_UpdateGateways *Keywords:*
Detail:	*Created on 28.06.2012. Last modified on 29.06.2012.*
GUID:	{69ED4217-A612-437d-8D5B-501CE0B83A31}

9.3 User Actions

Type: **Package**
Status: Reworked. Version 1.0. Phase 1.0.
Package: Control Interface
Detail: *Created on 09.09.2011. Last modified on 07.10.2011*
GUID: {737C575C-B3D0-48b2-A6AC-30DC12BB45D1}

user_Connection

Type: **Package**
Status: Reworked. Version 1.0. Phase 1.0.
Package: User Actions
Detail: *Created on 16.08.2011. Last modified on 07.10.2011*
GUID: {CCDEC92E-3CEF-4b6b-AC23-2A8BAE67CC5F}

user_Connection - *(Interaction diagram)*

Created By: christian *on* 16.08.2011
Last Modified: 17.10.2011
Version: 1.0. *Locked:* False
GUID: {D3A646D3-0224-4b0f-94C1-B4BF752BC61D}
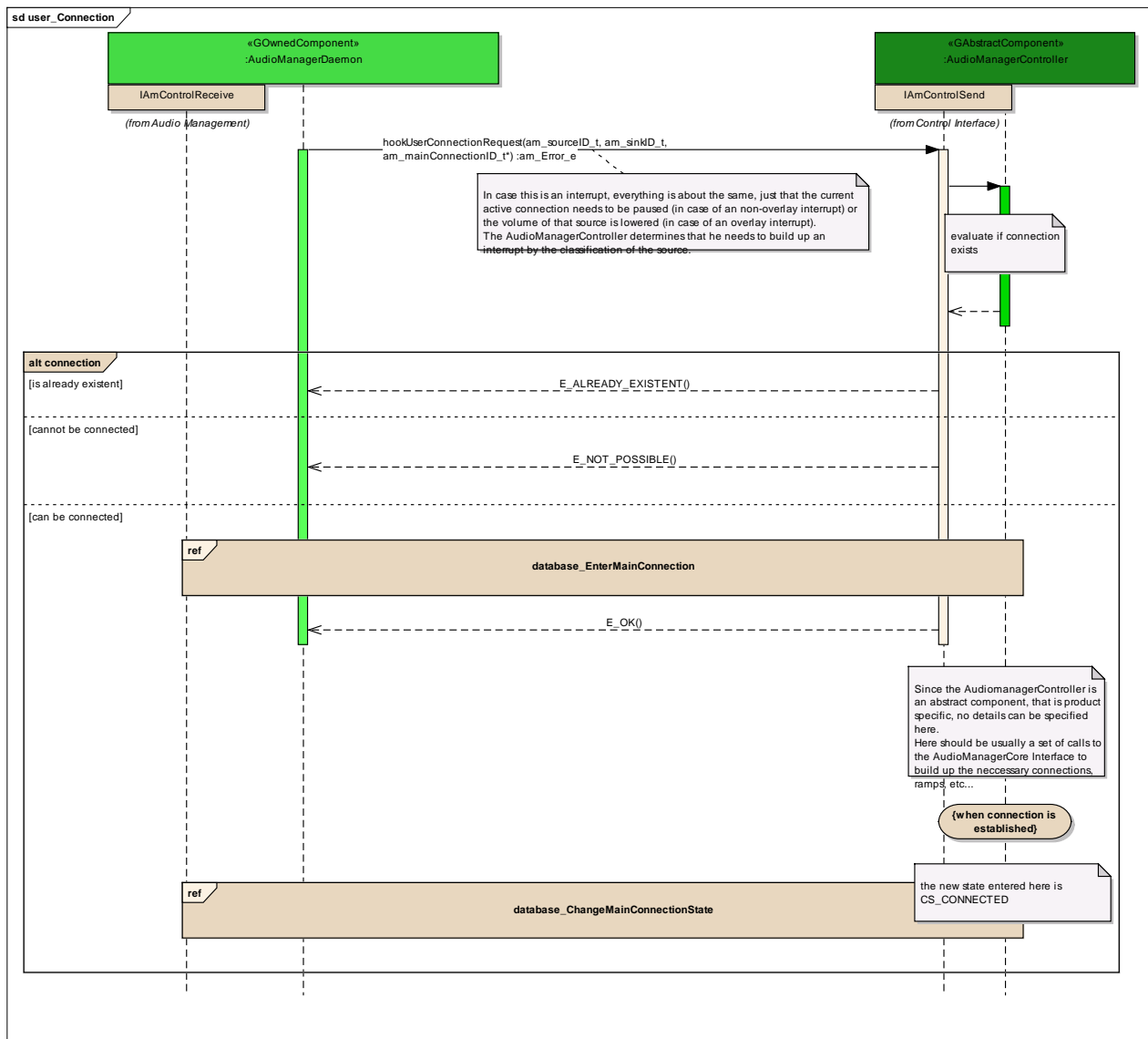
Figure: 76

Product specific

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_Connection *Keywords:*
Detail:	*Created on 17.08.2011. Last modified on 22.09.2011.*
GUID:	{781EC822-F8DD-4406-9187-E0CC6661B181}

connection

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_Connection *Keywords:*
Detail:	*Created on 17.08.2011. Last modified on 17.08.2011.*
GUID:	{A650CEA3-D7F7-4845-A4D8-7B5490E840C3}

database_ChangeMainConnectionState

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_Connection *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{4630E7BD-E406-4a0f-BF9A-F1DADF7F85CA}

database_EnterMainConnection

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_Connection *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{1491FE3B-5588-4d2c-A484-B072035BE81D}

user_Disconnection

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	User Actions
Detail:	*Created on 17.08.2011. Last modified on 07.10.2011*
GUID:	{508CB58E-5D1C-4a33-91EE-A586131E4BC5}

user_Disconnection - *(Interaction diagram)*

Created By:	christian *on* 17.08.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{538504BF-075F-4950-B28C-32B4F289906F}
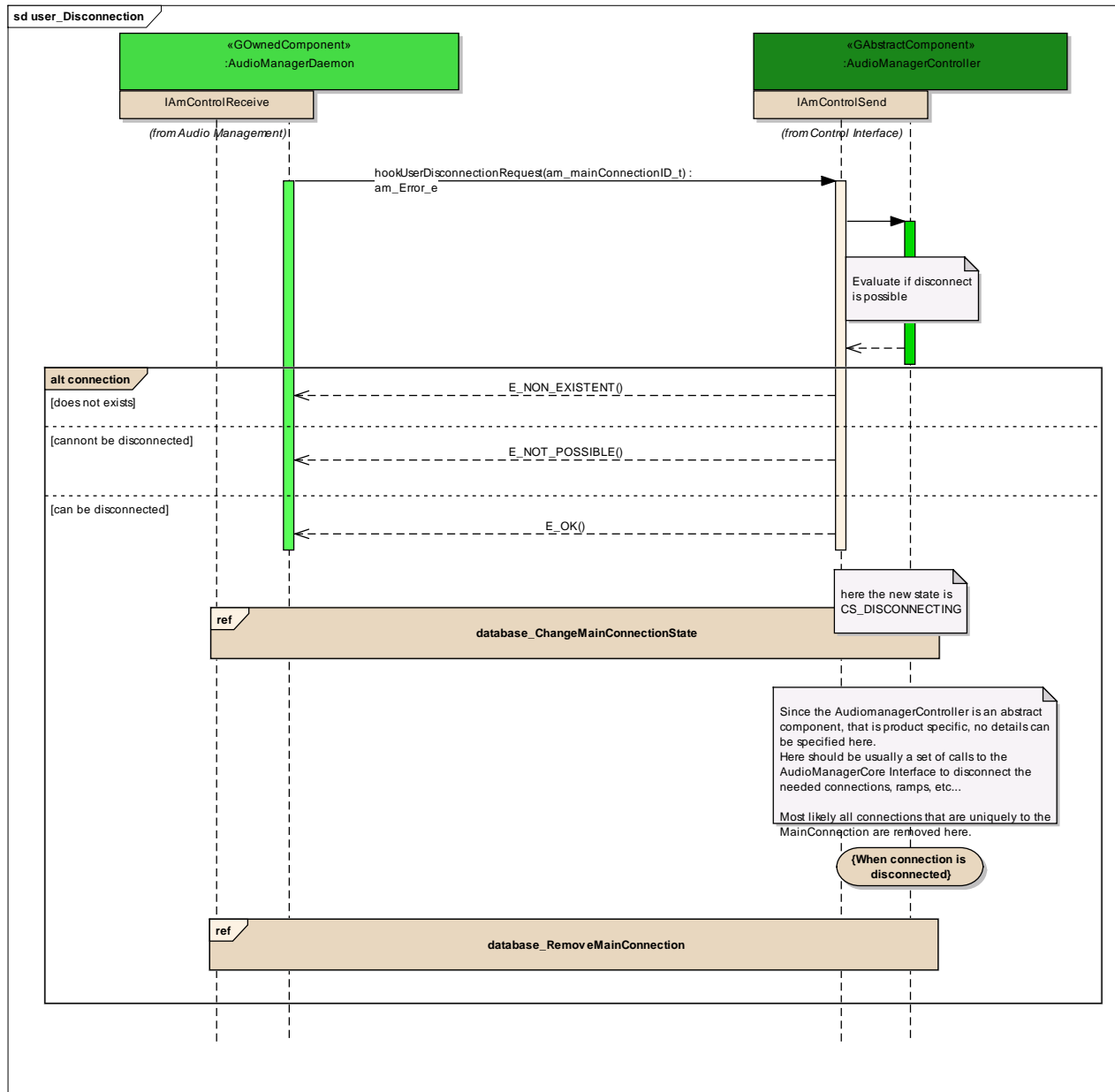
Figure: 77

Product specific

Type: **InteractionFragment**

Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_Disconnection *Keywords:*
Detail:	Created on 17.08.2011. Last modified on 17.08.2011.
GUID:	{139731F7-3AB3-4ac5-878E-1B325ED5D34B}

connection

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_Disconnection *Keywords:*
Detail:	Created on 17.08.2011. Last modified on 17.08.2011.
GUID:	{18AF9A6C-F655-4e8a-A054-907543758FBF}

database_ChangeMainConnectionState

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_Disconnection *Keywords:*
Detail:	Created on 10.10.2011. Last modified on 10.10.2011.
GUID:	{0A87A020-478E-40b7-8134-3694C980A7C9}

database_RemoveMainConnection

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_Disconnection *Keywords:*
Detail:	Created on 10.10.2011. Last modified on 10.10.2011.
GUID:	{36F693F7-B05B-4c12-BBFB-FE1294DE36A0}

user_MuteStateChange

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	User Actions
Detail:	*Created on 22.09.2011. Last modified on 07.10.2011*
GUID:	{B687F3AF-46F2-448f-BA21-AC2E4C4BD852}

user_MuteStateChange - *(Interaction diagram)*

Created By:	christian *on* 22.09.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{D8FAF9DE-F511-445e-B49A-96155F51C38A}
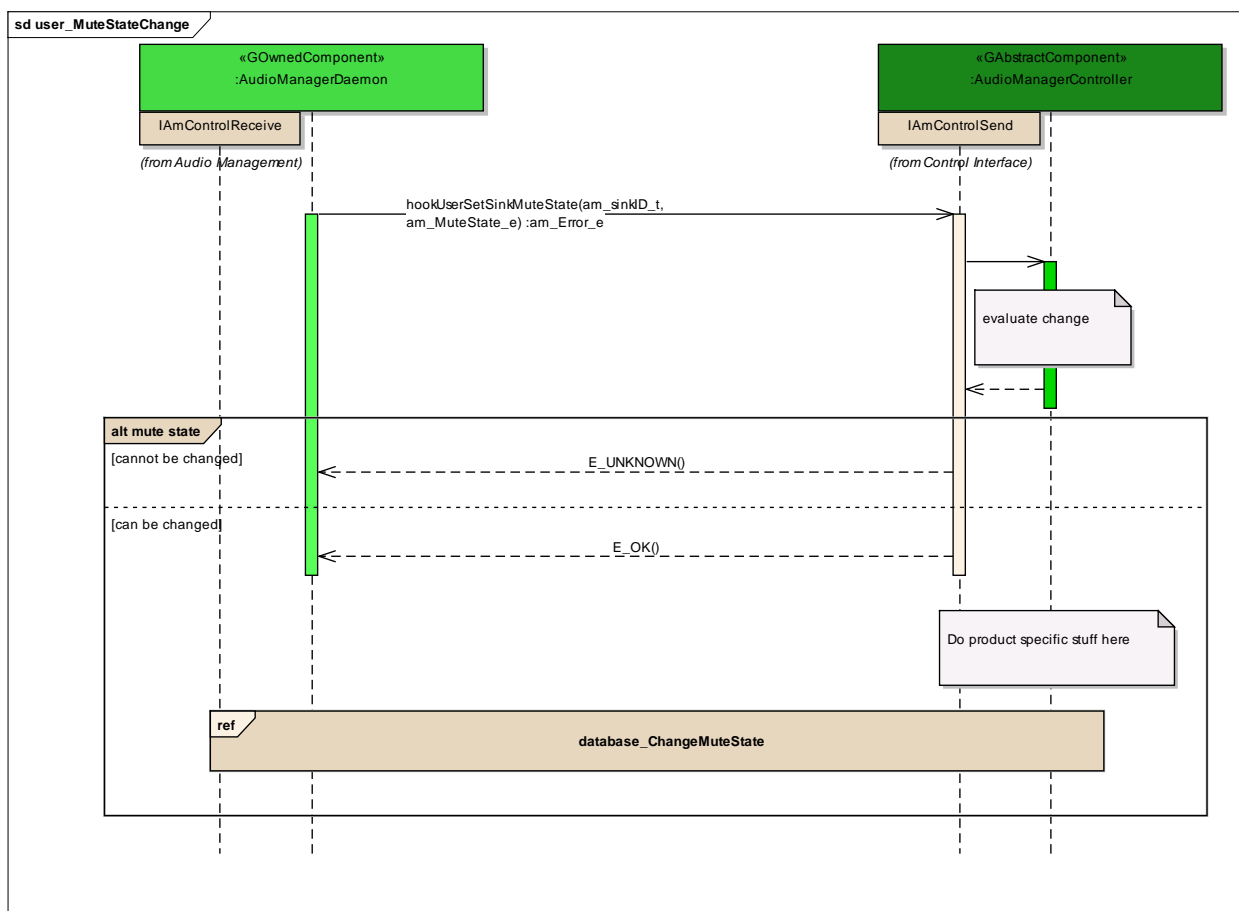
Figure: 78

database_ChangeMuteState

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.

Package:	user_MuteStateChange	*Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*	
GUID:	{88CEAFAC-318E-4a0d-972D-8E3013C24704}	

mute state

Type:	**InteractionFragment**	
Status:	Proposed. Version 1.0. Phase 1.0.	
Package:	user_MuteStateChange	*Keywords:*
Detail:	*Created on 22.09.2011. Last modified on 22.09.2011.*	
GUID:	{F7651FED-4D3B-41b6-A6C1-0BC3FE0ACDF0}	

product specific

Type:	**InteractionFragment**	
Status:	Proposed. Version 1.0. Phase 1.0.	
Package:	user_MuteStateChange	*Keywords:*
Detail:	*Created on 22.09.2011. Last modified on 22.09.2011.*	
GUID:	{F6FAEF7E-49E5-485b-AACB-57093F5D4C58}	

user_SinkSoundProperty Change

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	User Actions
Detail:	*Created on 12.09.2011. Last modified on 07.10.2011*
GUID:	{6F3868C2-7097-4c27-8690-7CA1A2ACD447}

user_SinkSoundProperty Change - *(Interaction diagram)*

Created By:	christian *on* 12.09.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
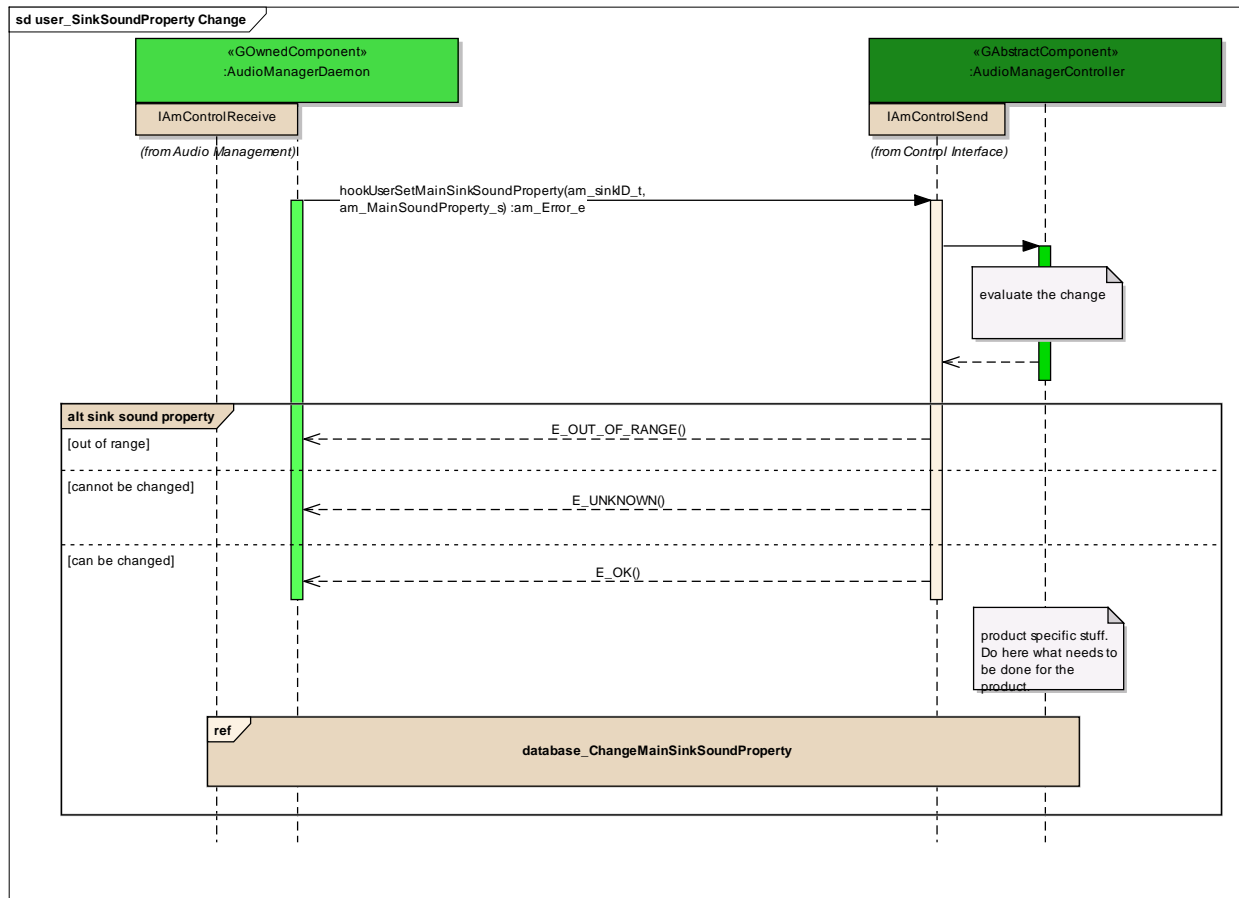
Figure: 79

database_ChangeMainSinkSoundProperty

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: user_SinkSoundProperty Change   *Keywords:*
Detail: *Created on 10.10.2011. Last modified on 10.10.2011.*
GUID: {246AE829-71B9-4991-80DE-D26424EA2045}

sink sound property

Type: **InteractionFragment**
Status: Proposed. Version 1.0. Phase 1.0.

Package:	user_SinkSoundProperty Change	*Keywords:*
Detail:	*Created on 19.09.2011. Last modified on 19.09.2011.*	
GUID:	{46B6A3B2-BC58-42e6-A55C-1E76F21E434D}	

user_SourceSoundPropertyChange

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	User Actions
Detail:	*Created on 12.09.2011. Last modified on 07.10.2011*
GUID:	{C579220E-F717-4e26-BB8C-22F5E2BB38E5}

user_SourceSoundPropertyChange - *(Interaction diagram)*

Created By:	christian *on* 12.09.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{31CC48B4-FFB8-493f-9D8B-338D7F29C47C}
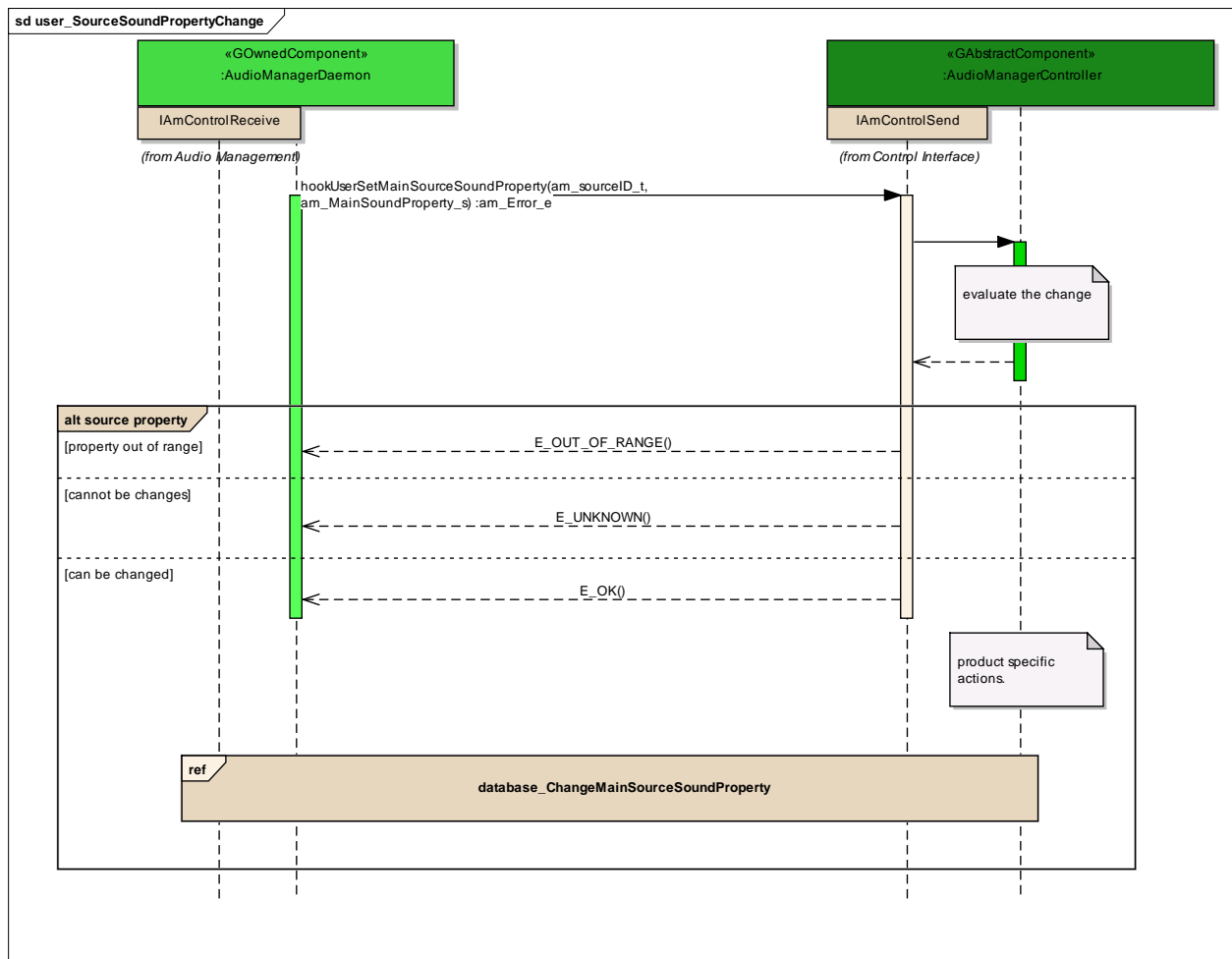
Figure: 80

database_ChangeMainSourceSoundProperty

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_SourceSoundPropertyChange *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{EB7A40A9-7937-42fb-BA66-4F4F5E008EB3}

source property

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.

Package:	user_SourceSoundPropertyChange	*Keywords:*
Detail:	*Created on 19.09.2011. Last modified on 19.09.2011.*	
GUID:	{255B51F8-40C4-4b0b-B0F5-F32482E54E6F}	

user_SystemPropertyChange

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	User Actions
Detail:	*Created on 06.10.2011. Last modified on 07.10.2011*
GUID:	{4D2AEB42-9C95-44a9-A923-BD6DBC90261A}

user_SystemPropertyChange - *(Interaction diagram)*

Created By:	christian *on* 06.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{4EA44395-71F2-454e-BC5E-E0D936A1DA56}
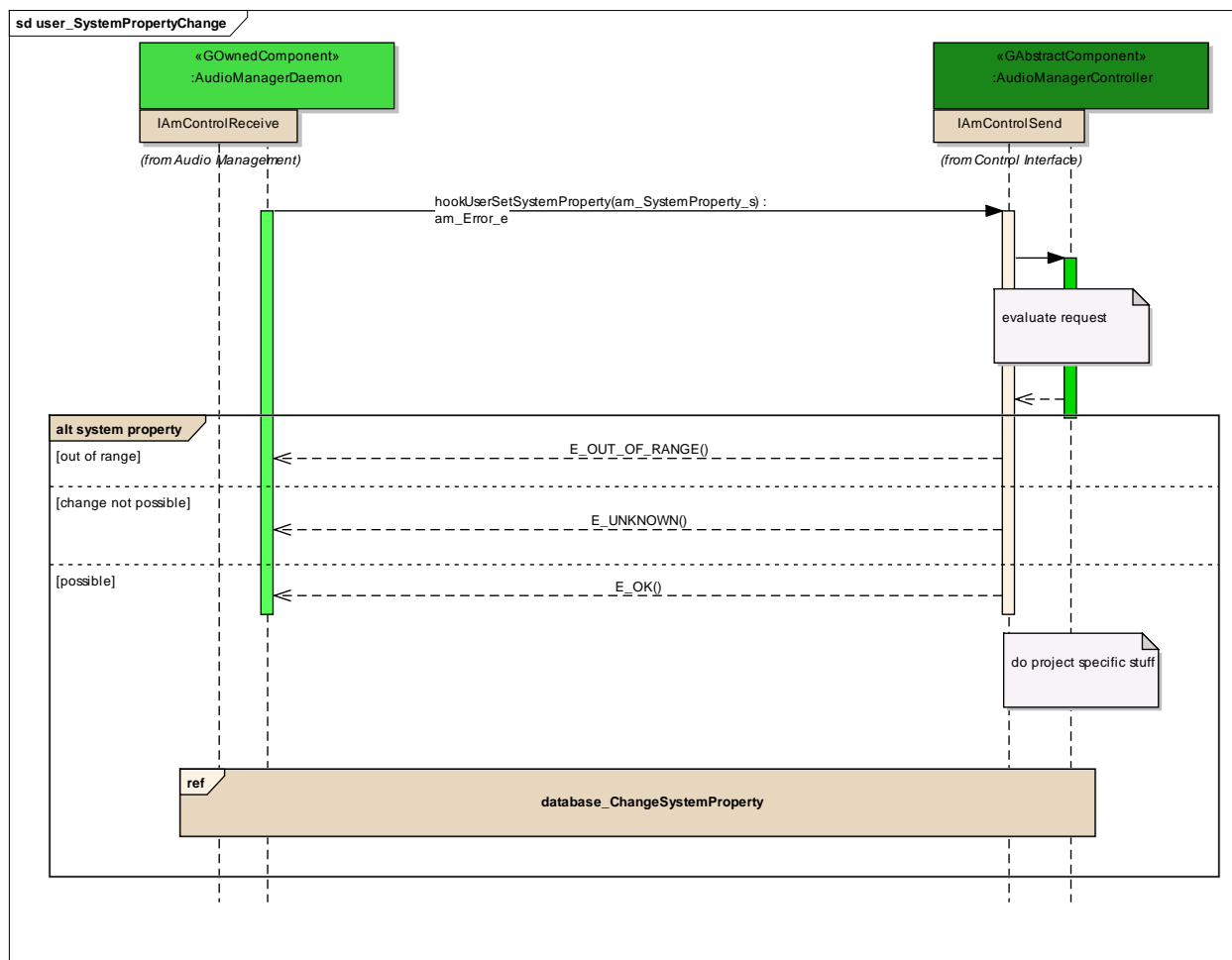
Figure: 81

database_ChangeSystemProperty

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_SystemPropertyChange *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{419784A3-5DCB-4ac3-BE8A-D626CF640F81}

system property

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_SystemPropertyChange *Keywords:*

Detail: *Created on 06.10.2011. Last modified on 06.10.2011.*
GUID: {B82FD946-3ADE-4e97-AD0E-CE1BC32B5018}

user_VolumeStep

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	User Actions
Detail:	*Created on 06.10.2011. Last modified on 10.10.2011*
GUID:	{0BB693F5-9569-4219-B7DC-B08FA48A9797}

user_VolumeStep - *(Interaction diagram)*

Created By:	christian *on* 06.10.2011
Last Modified:	10.10.2011
Version:	1.0. *Locked:* False
GUID:	{48D679B5-41E2-4f57-83CF-C8ED13E6B9D7}
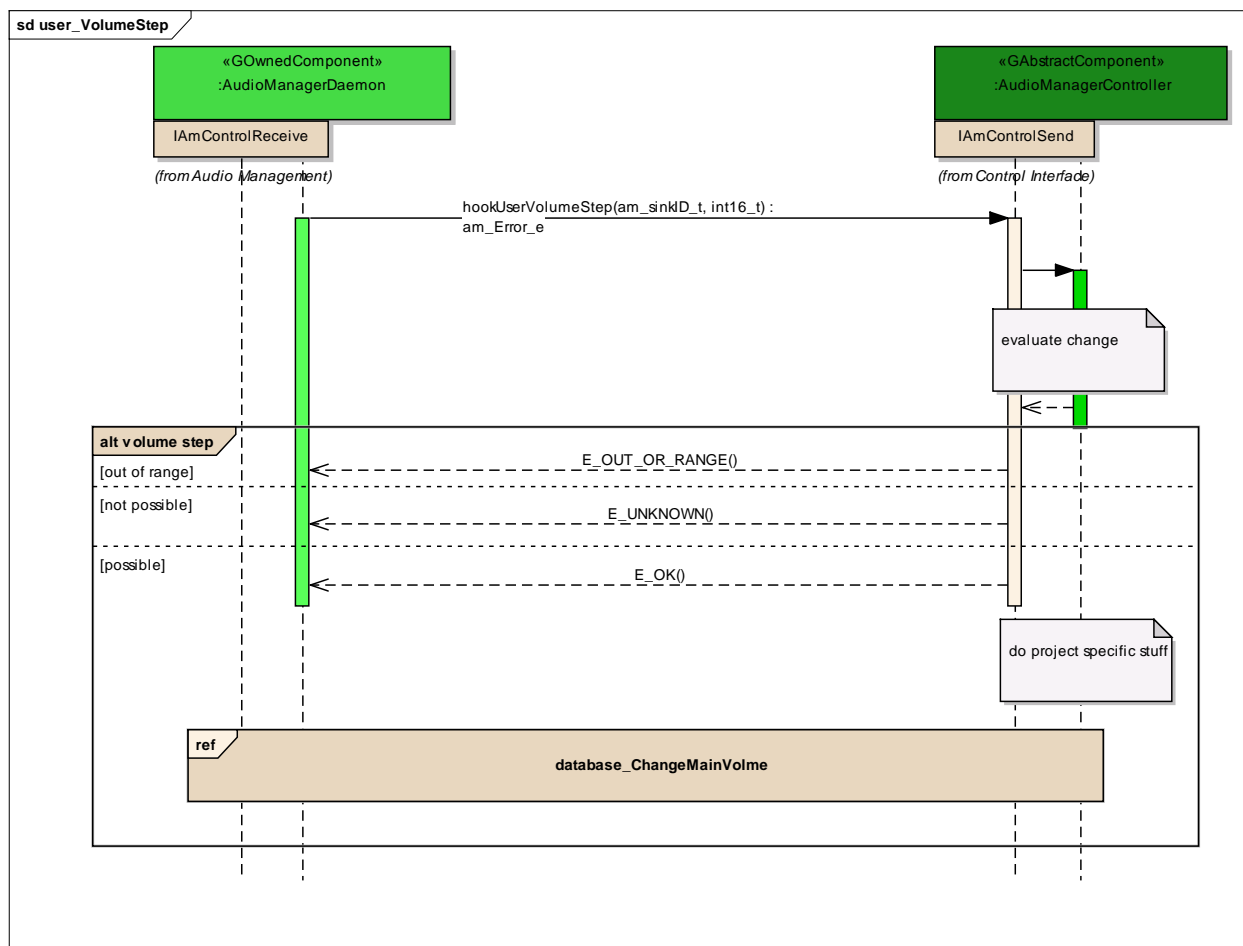
Figure: 82

database_ChangeMainVolme

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_VolumeStep *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{8C174B5B-4CAE-4f2e-BF41-36B706C6A002}

volume step

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_VolumeStep *Keywords:*

Detail: *Created on 06.10.2011. Last modified on 06.10.2011.*
GUID: {BAECDE87-D6A1-467c-9DB7-00B1D89F31A2}

user_VolumeChange

Type: **Package**
Status: Reworked. Version 1.0. Phase 1.0.
Package: User Actions
Detail: *Created on 17.08.2011. Last modified on 10.10.2011*
GUID: {338615B8-9305-4fe2-B8B1-6CA82D4B32DF}

user_VolumeChange - *(Interaction diagram)*
Created By: christian *on* 17.08.2011
Last Modified: 10.10.2011
Version: 1.0. *Locked:* False
GUID: {36BE8A26-3177-4c16-951F-33BB26EFA6EE}

sd user_VolumeChange

«GOwnedComponent»
:AudioManagerDaemon

IAmControlReceive

(from Audio Management)

«GAbstractComponent»
:AudioManagerController

IAmControlSend

(from Control Interface)

hookUserVolumeChange(am_sinkID_t, am_mainVolume_t)
:am_Error_e

evaluate volume change

alt new volume

[is out of range]

E_OUT_OF_RANGE()

[can be set]

E_OK()

Now several sequences could be implemented. For example changing the volume at the source or changing it at the sink - but this is product specific.

{When volume was changed}

ref

database_ChangeMainVolme

Since the implementation of the AudioMangerController is product specific, this cannot be the real implementation.
It is just a suggestion to demonstrate the sequence flow.

database_ChangeMainVolme

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_VolumeChange *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{5966DA05-FEB5-4f71-9F48-634B69EB1389}

new volume

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	user_VolumeChange *Keywords:*
Detail:	*Created on 18.08.2011. Last modified on 18.08.2011.*
GUID:	{F3E16D7C-7802-4d79-8217-56983F3AF3FE}

9.4 System Events

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	Control Interface
Detail:	*Created on 09.09.2011. Last modified on 10.10.2011*
GUID:	{084835B0-A0F8-4d14-8E75-917D49150329}

System

Type:	**Component**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	System Events *Keywords:*
Detail:	*Created on 22.09.2011. Last modified on 22.09.2011.*
GUID:	{D24AEC54-BCCE-4711-835F-FC9B86DD6B2C}

- isIndirectlyInstantiated = True

Embedded Elements

Element	Detail	Notes
ProvidedInterface Vehicle speed	Version: 1.0	

events_MainSinkNotificationConfigurationChange

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013*
GUID:	{F4F5863F-3236-48eb-BA07-12D409353AD5}

events_MainSinkNotificationConfigurationChange - *(Interaction diagram)*

Created By:	q232968 *on* 29.01.2013
Last Modified:	29.01.2013
Version:	1.0. *Locked:* False
GUID:	{D29DA93E-4B9E-4e56-A498-597B289CA1AB}
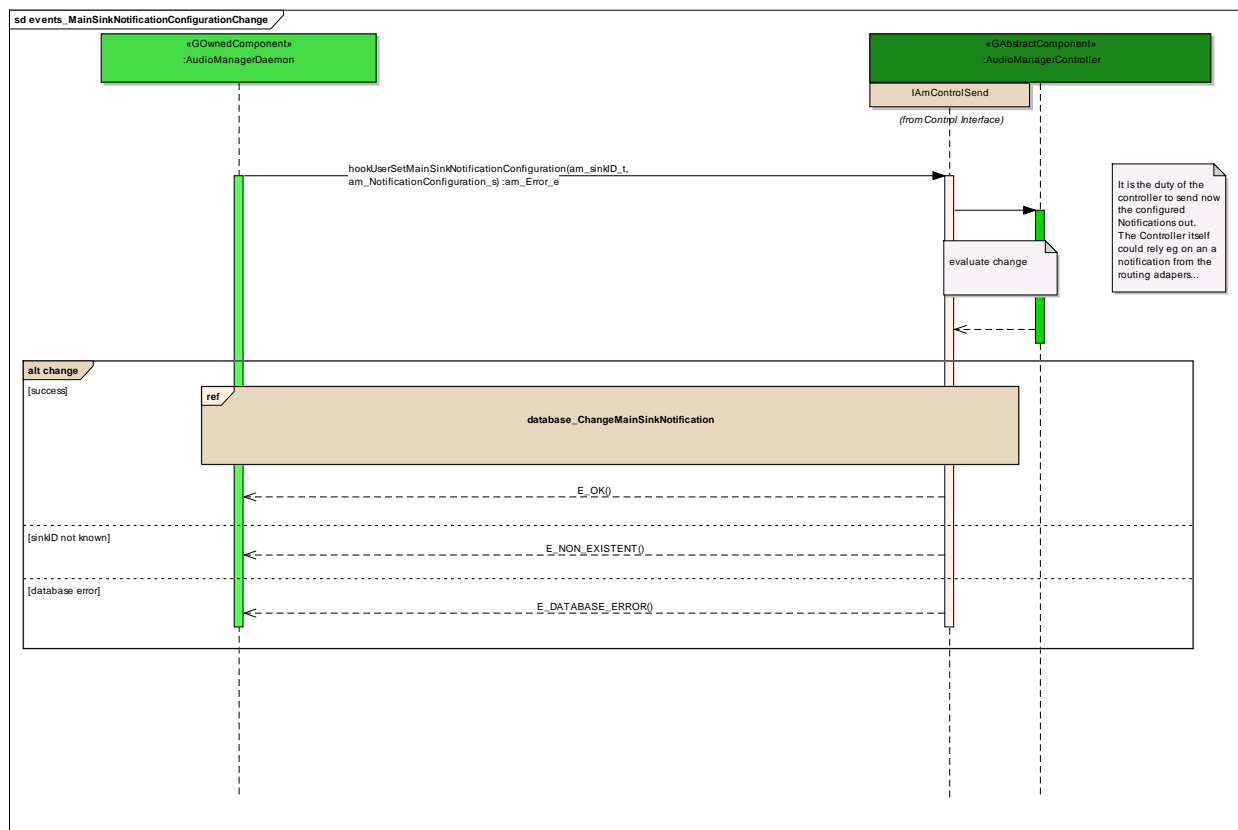
Figure: 84

change

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_MainSinkNotificationConfigurationChange *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{AC590DA6-9D45-4b46-801D-E00727B765A9}

database_ChangeMainSinkNotification

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_MainSinkNotificationConfigurationChange *Keywords:*
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013.*
GUID:	{A6578DDA-D4FE-46c4-BA0B-E331E1C56BF2}

events_MainSourceNotficiationConfigurationChange

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 29.01.2013. Last modified on 29.01.2013*
GUID:	{01557D15-CC43-473b-AB95-B148A2DE1B73}

events_MainSourceNotficiationConfigurationChange - *(Interaction diagram)*

Created By:	q232968 *on* 29.01.2013
Last Modified:	29.01.2013
Version:	1.0. *Locked:* False
GUID:	{B3521A43-5356-4f8e-96DF-7872960228A6}
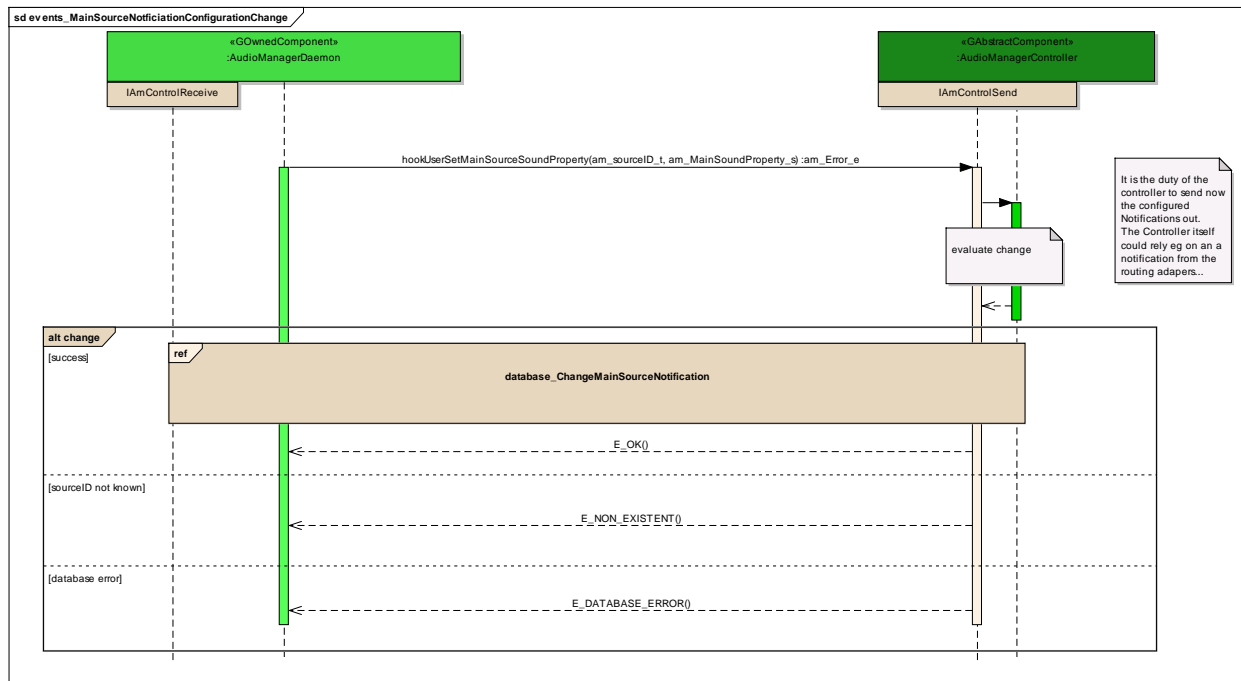
Figure: 85

change

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.

database_ChangeMainSourceNotification

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_MainSourceNotficiationConfigurationChange   *Keywords:*
Detail: *Created on 29.01.2013. Last modified on 29.01.2013.*
GUID: {A7D7D00C-71D4-4118-9D1E-3927AC6531B9}

events_SinkNotificationDataChange

Type: **Package**
Status: Proposed. Version 1.0. Phase 1.0.
Package: System Events
Detail: *Created on 29.01.2013. Last modified on 29.01.2013*
GUID: {585AD876-0757-4269-8634-B9AA2C95C720}

events_SinkNotificationDataChange - *(Interaction diagram)*

Created By: q232968 *on* 29.01.2013
Last Modified: 30.01.2013
Version: 1.0. *Locked:* False
GUID: {972E26CC-D87F-4166-B9C4-FE392FAB7E9A}
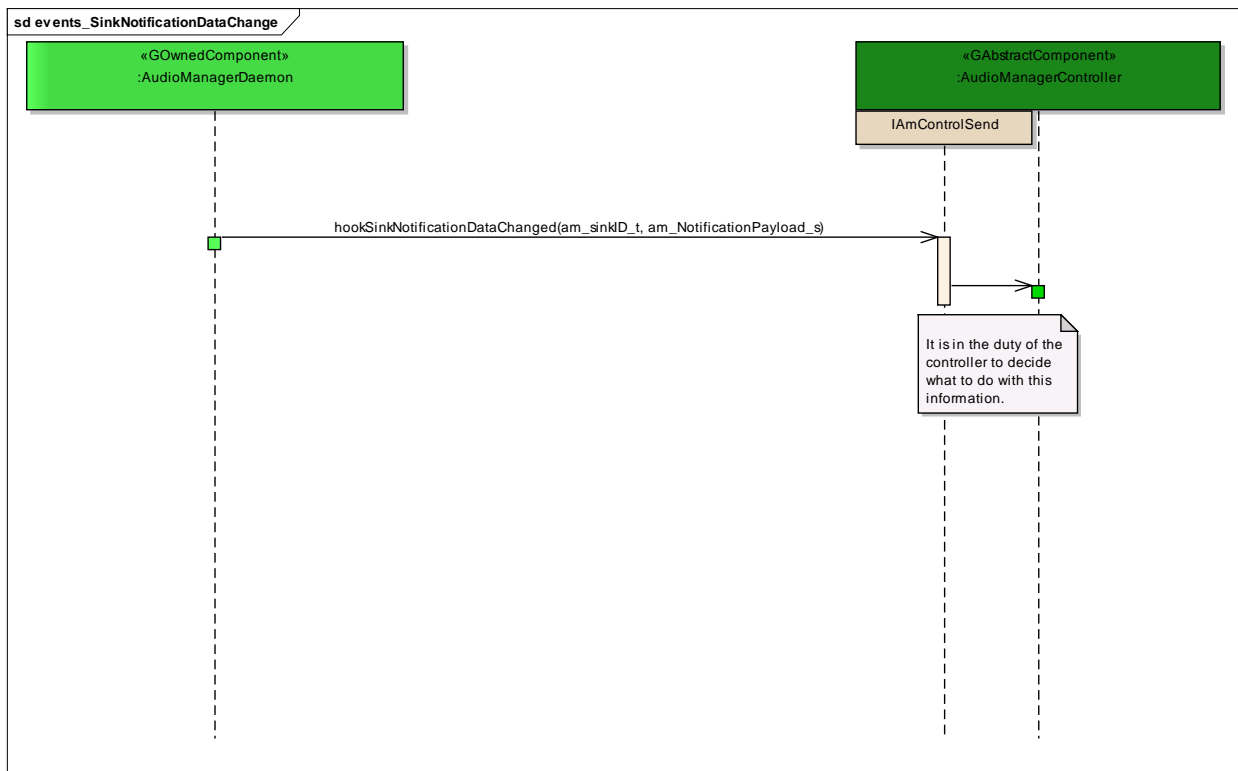
Figure: 86

events_SourceNotificationDataChange

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 30.01.2013. Last modified on 30.01.2013*
GUID:	{07856D62-9368-46ea-8D7D-A1CD5225ABF8}

events_SourceNotificationDataChange - *(Interaction diagram)*

Created By:	q232968 *on* 30.01.2013
Last Modified:	30.01.2013
Version:	1.0. *Locked:* False
GUID:	{484E32A8-DE4D-48d6-B998-59EF74BCC0D3}

Figure: 87

events_AllPluginsLoaded

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011*
GUID:	{E8E18AA2-53E9-496d-AD04-83BA800FC2EE}

events_AllPluginsLoaded - *(Interaction diagram)*

Created By:	christian *on* 12.10.2011
Last Modified:	07.03.2012
Version:	1.0. *Locked:* False
GUID:	{6189C7C4-08EE-4511-A76A-149B3C476BAE}
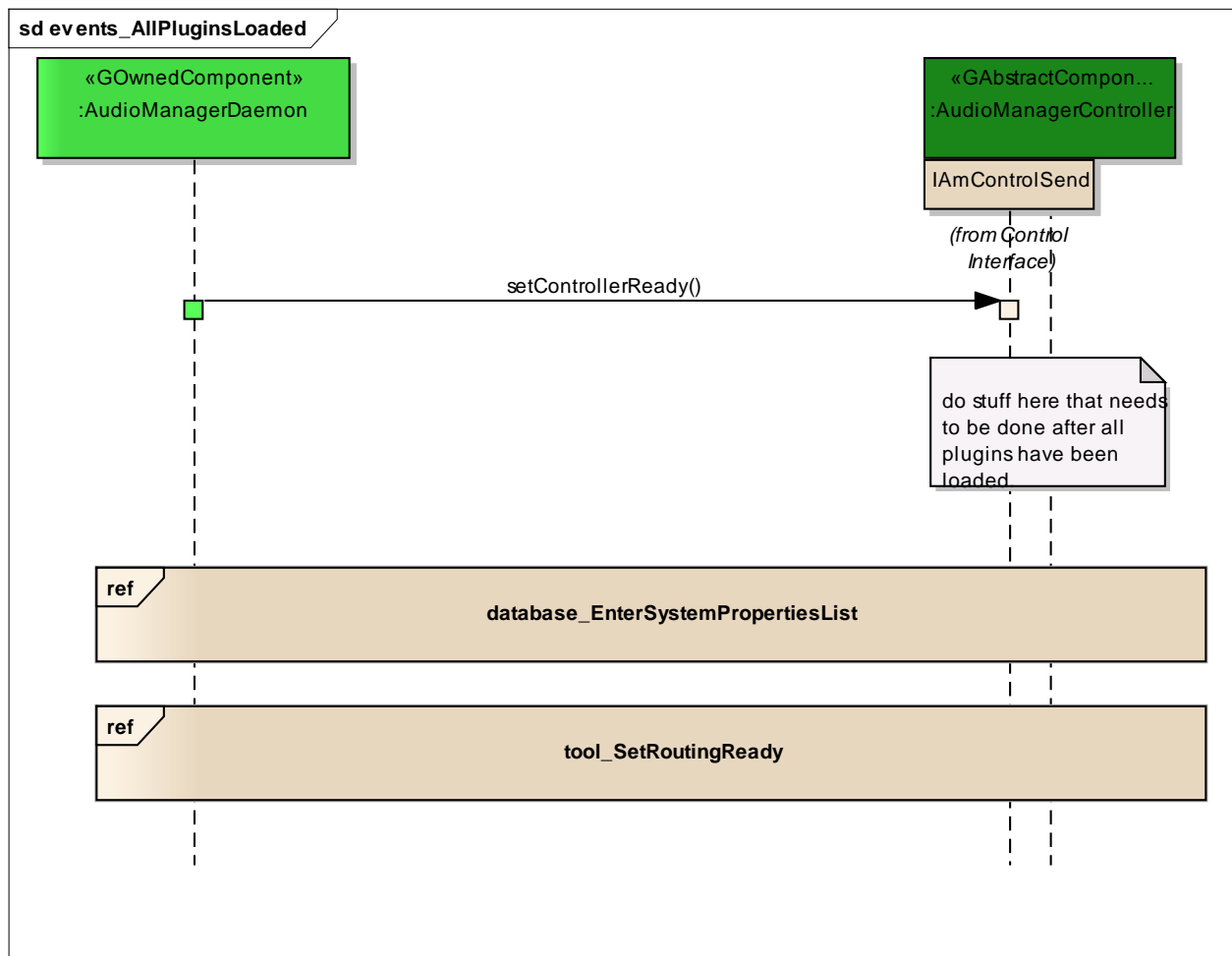
Figure: 88

database_EnterSystemPropertiesList

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_AllPluginsLoaded   *Keywords:*
Detail: *Created on 07.12.2011. Last modified on 07.12.2011.*
GUID: {3CB5F697-CFDE-4c2e-8A78-39D05CD83AF3}

tool_SetRoutingReady

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.

Package:	events_AllPluginsLoaded *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{7B438642-29A3-4d3a-B747-5D345FE83C4E}

events_DomainRegistrationComplete

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011*
GUID:	{55867179-3D38-4109-8702-8386E1393AA5}

events_DomainRegistrationComplete - *(Interaction diagram)*

Created By:	christian *on* 12.10.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{0ED3963B-2A68-4cc1-BEC4-A82D2067357C}
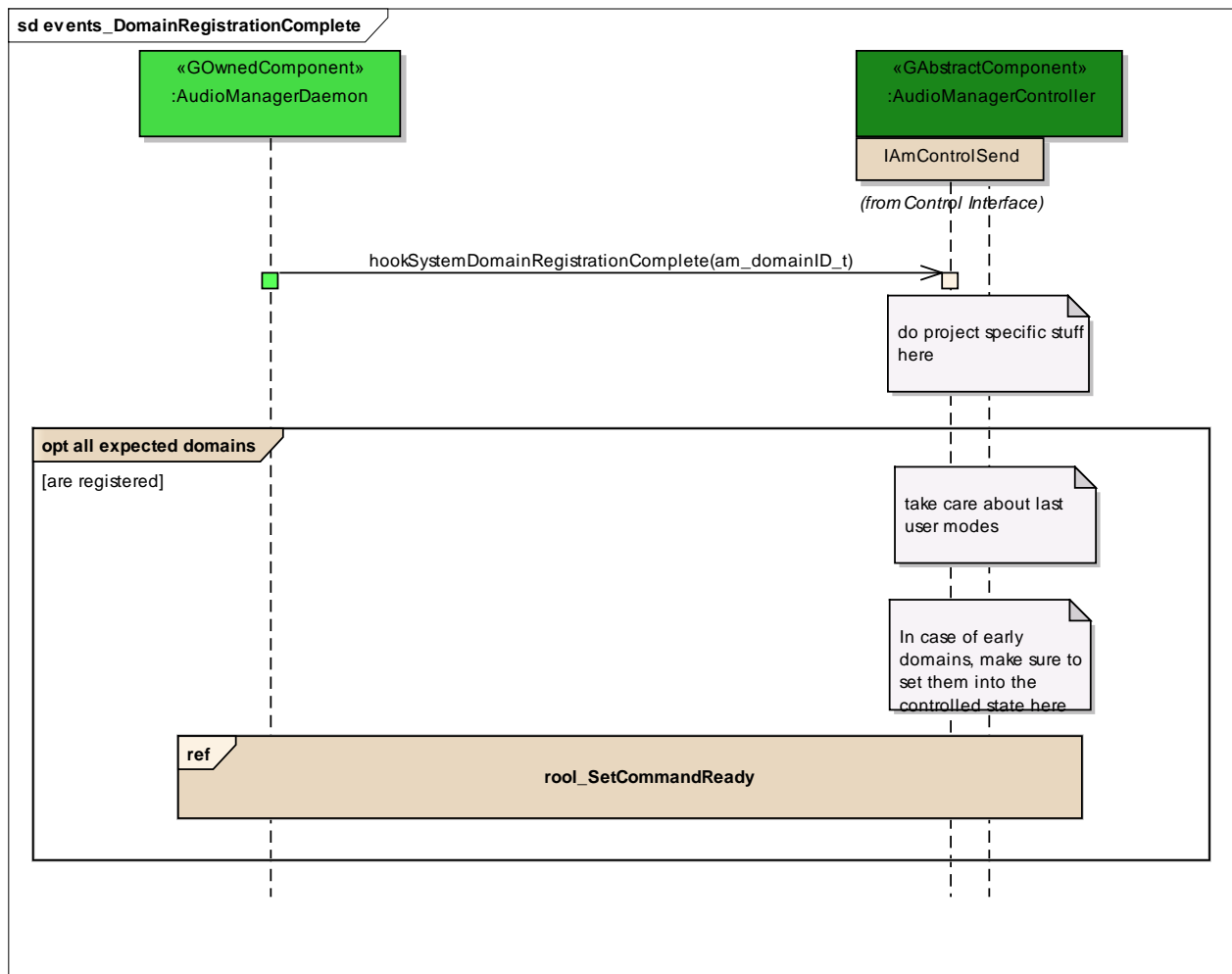
Figure: 89

all expected domains

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainRegistrationComplete *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{21ACDE05-78B3-4967-B356-9FDE4A56FE7C}

rool_SetCommandReady

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.

Package:	events_DomainRegistrationComplete *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{CE5C3FBA-4FA9-48a5-A143-C3F21CB524BF}

events_SinkVolumeTicks

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 04.10.2011. Last modified on 11.10.2011*
GUID:	{49840983-DF25-4101-968F-D5025992A681}

events_SinkVolumeTicks - *(Interaction diagram)*

Created By:	christian *on* 04.10.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{2C6E09DC-95B7-43bc-9CBE-4A8A9C1B150C}
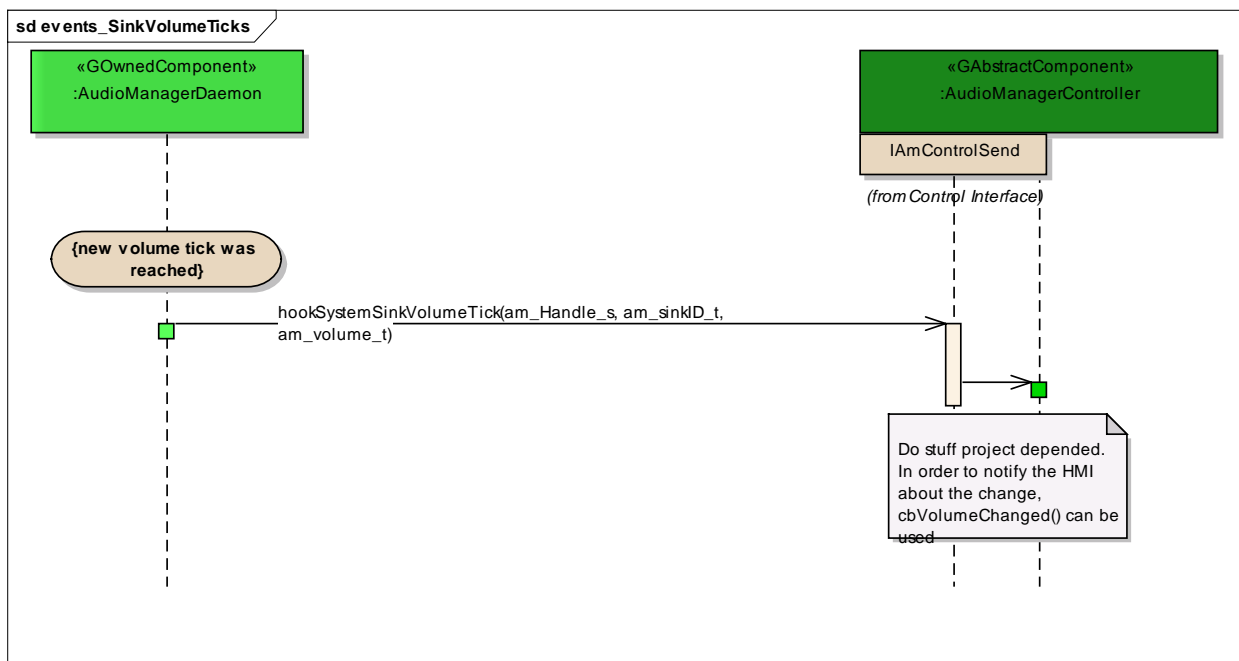
Figure: 90

events_SourceVolumeTicks

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 04.10.2011. Last modified on 11.10.2011*
GUID:	{FE297D2F-D849-4b2b-AB35-2CD273F15138}

events_SourceVolumeTicks - *(Interaction diagram)*

Created By:	christian *on* 04.10.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{817DD251-1618-4b1e-B31C-E1C528415677}

Figure: 91

events_DomainStateChange

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 21.09.2011. Last modified on 11.10.2011*
GUID:	{28EB459A-7418-4326-A673-33EDFA48D4FE}

events_DomainStateChange - *(Interaction diagram)*

Created By:	christian *on* 21.09.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{46682A21-E50D-48ea-ABBA-AE1857844504}
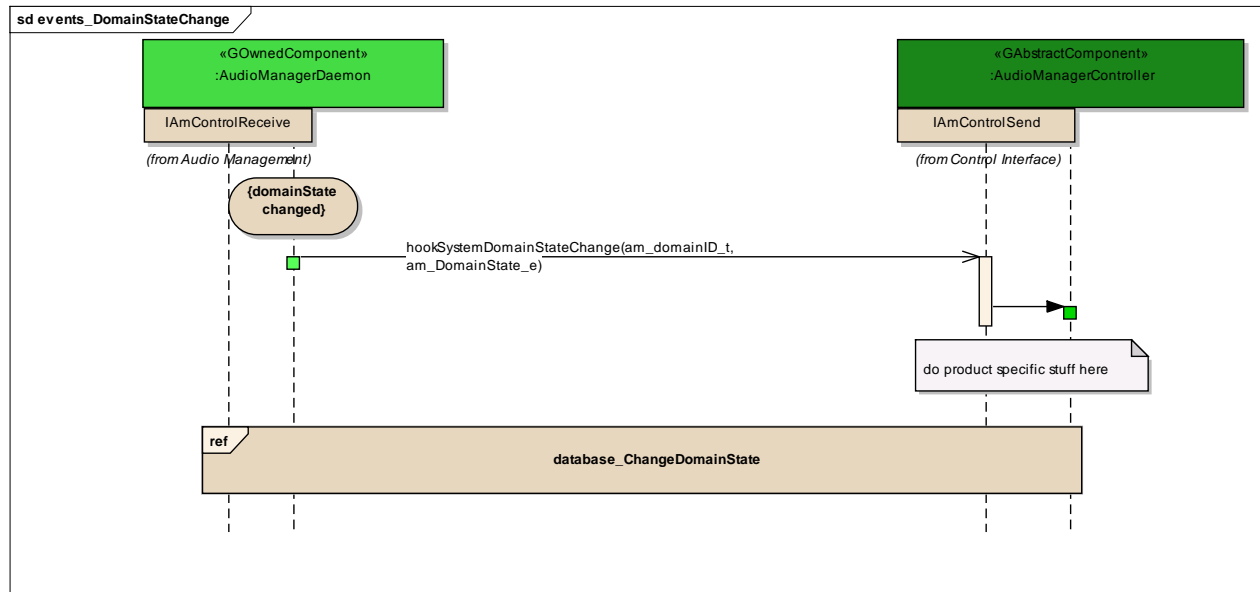
Figure: 92

database_ChangeDomainState

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainStateChange *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{A7CD06A0-B30F-47b7-A7CC-01DEB7589169}

events_SinkAvailabilityChange

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 20.09.2011. Last modified on 11.10.2011*
GUID:	{CD245B6E-ACE1-49a7-9327-B949152394D7}

events_SinkAvailabilityChange - *(Interaction diagram)*

Created By:	christian *on* 20.09.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{69F3C507-BD00-46bd-B010-1C206E83E779}

Figure: 93

database_ChangeSinkAvailability

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SinkAvailabilityChange *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{7F8149EC-8F2F-453a-B5F9-7717CD009195}

events_SourceAvailabilityChange

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 19.09.2011. Last modified on 11.10.2011*

events_SourceAvailabilityChange - *(Interaction diagram)*
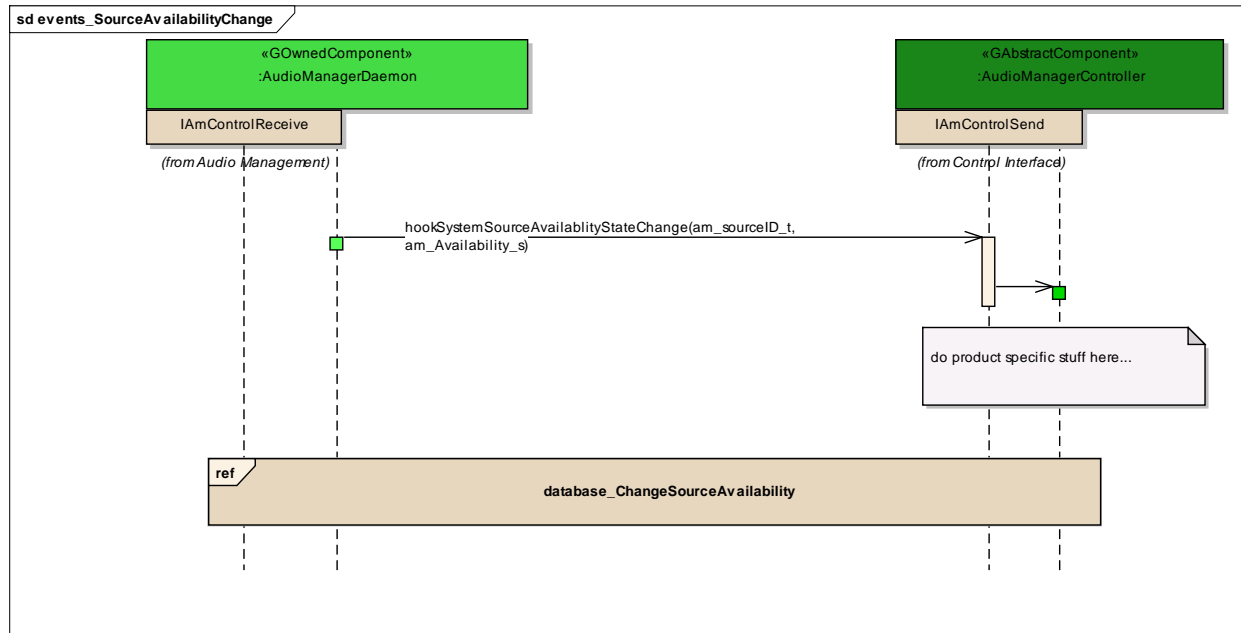
Figure: 94

c_Source_Availability_Change

database_ChangeSourceAvailability

Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{5D3C5FEC-41F8-41fa-858E-B12917CFE97B}

product specific

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SourceAvailabilityChange *Keywords:*
Detail:	*Created on 19.09.2011. Last modified on 19.09.2011.*
GUID:	{960DFE8A-3180-416a-9DB6-E5012E8919A4}

events_SpeedChange

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 22.09.2011. Last modified on 11.10.2011*
GUID:	{B4DD4695-5E65-4943-A2B8-106BB2D9EA42}

The adjustments for the speed are product specific in the controller. Here is shown how the AudioManagerDaemon provides speed information with a given quantization and sample rate.

Turning speed controlled volume on/off and possible settings are achieved via SinkSoundProperty settings.

events_SpeedChange - *(Interaction diagram)*

Created By:	christian *on* 22.09.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{9775708F-6671-4d48-AAB0-53E5AAEBF7CB}
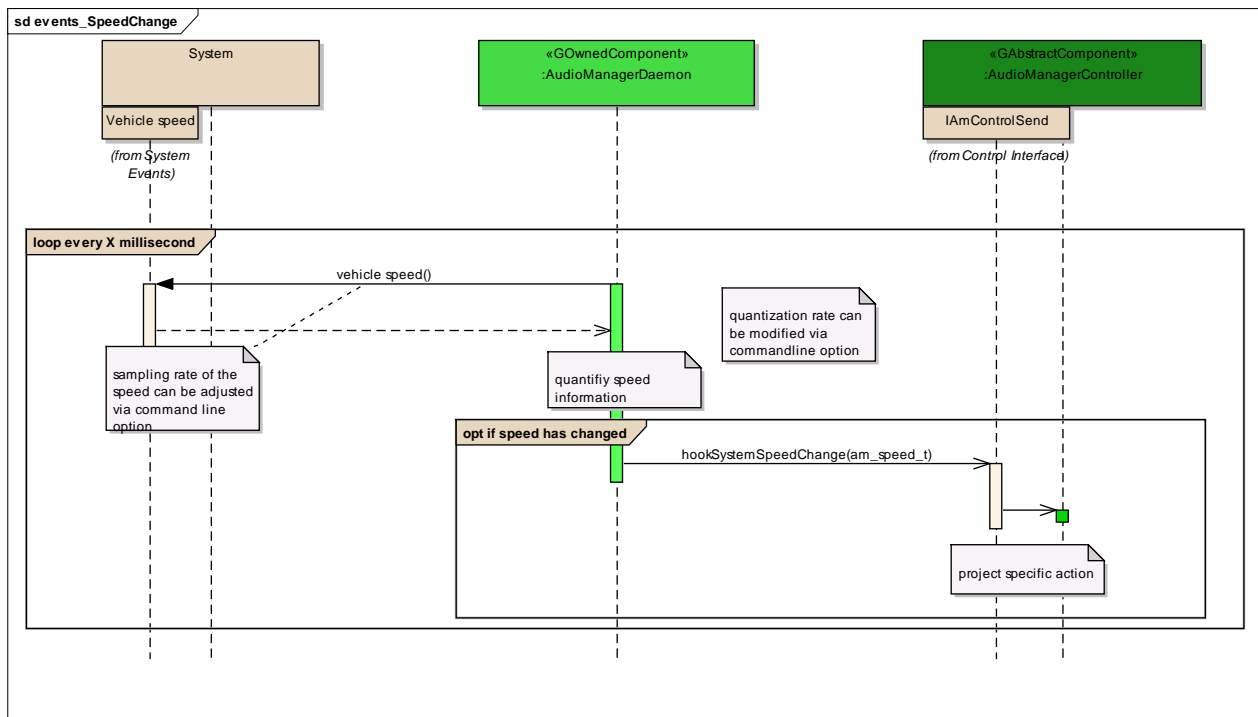
Figure: 95

every X millisecond

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SpeedChange *Keywords:*
Detail:	*Created on 22.09.2011. Last modified on 22.09.2011.*
GUID:	{0C42952F-7512-47f0-B709-9251D546BB10}

if speed has changed

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SpeedChange *Keywords:*
Detail:	*Created on 22.09.2011. Last modified on 22.09.2011.*
GUID:	{291554A4-54EF-456b-BB7B-D2BD36EC76BE}

events_SinkRegistration

Type: **Package**
Status: Reworked. Version 1.0. Phase 1.0.
Package: System Events
Detail: *Created on 22.08.2011. Last modified on 11.10.2011*
GUID: {C9C6DFA8-90A7-480c-8E0D-8096E8A88E38}

events_SinkRegistration - *(Interaction diagram)*

Created By: christian *on* 22.08.2011
Last Modified: 11.10.2011
Version: 1.0. *Locked:* False
GUID: {2BD71163-5726-45c2-9AE3-E0838C3C4EB7}

Figure: 96

database_EnterSink

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SinkRegistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{993FA127-4B7F-4c9c-8E4D-4E73F8C280C0}

sink registration

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SinkRegistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{D4D6FC47-B1F6-4f17-B328-FB78E8A0D67F}

events_SinkDeregistration

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 13.09.2011. Last modified on 11.10.2011*
GUID:	{D32F6CC1-24A6-4368-9A48-091EBA1538AD}

events_SinkDeregistration - *(Interaction diagram)*

Created By:	christian *on* 13.09.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{106DE032-11C9-4ead-9865-4BC4C6B015E5}
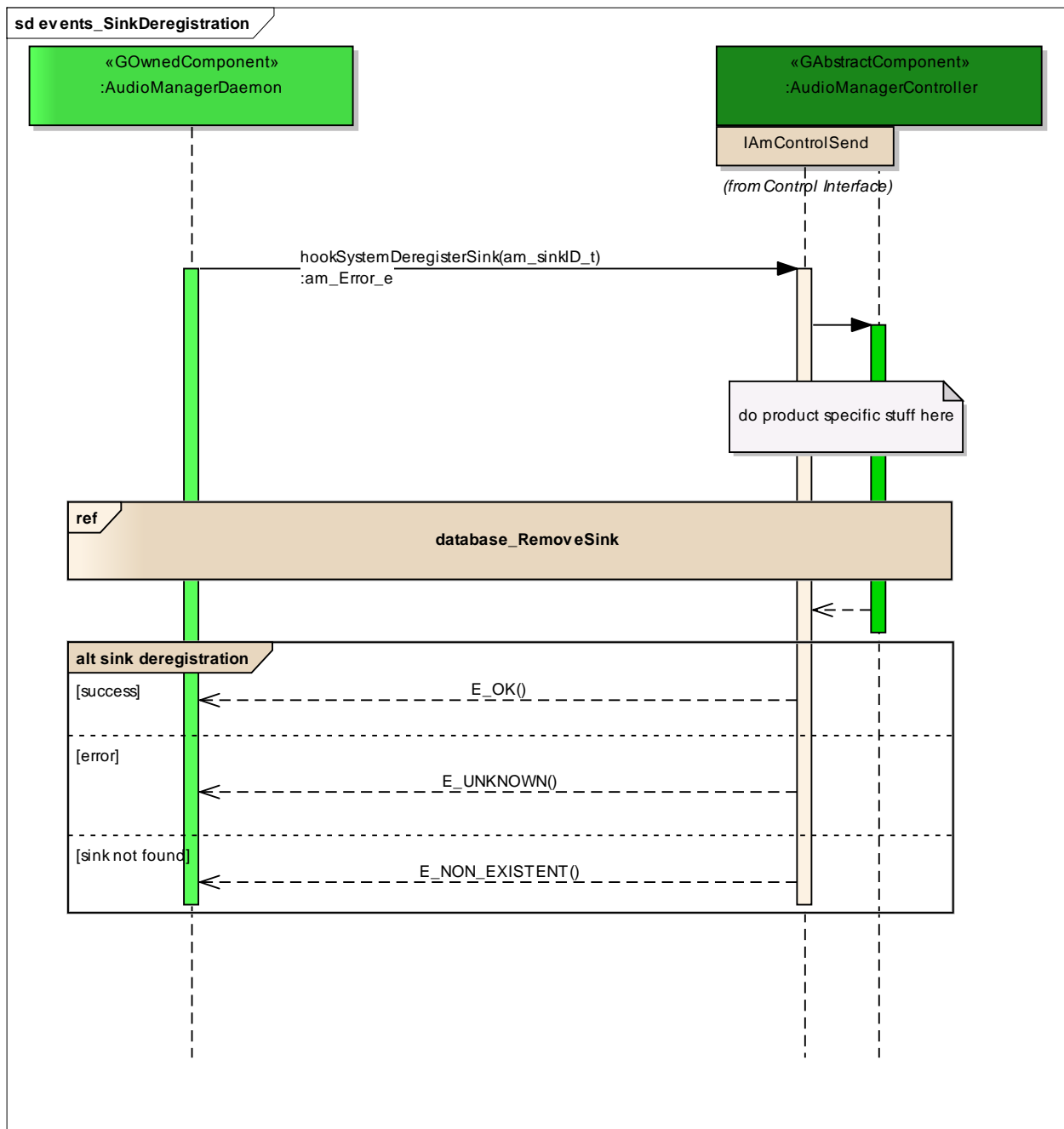
Figure: 97

database_RemoveSink

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SinkDeregistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{D5171584-7D09-4f6e-86D1-63C008541076}

sink deregistration

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SinkDeregistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{DADB8E0D-A637-44a3-ADBC-26847F0F3989}

events_SourceRegistration

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 22.08.2011. Last modified on 11.10.2011*
GUID:	{40186ABC-0DCF-40ee-BAE7-AAEAEF19FA9A}

events_SourceRegistration - *(Interaction diagram)*

Created By:	christian *on* 22.08.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{21ED773A-488C-4537-89DF-68CA2125E166}
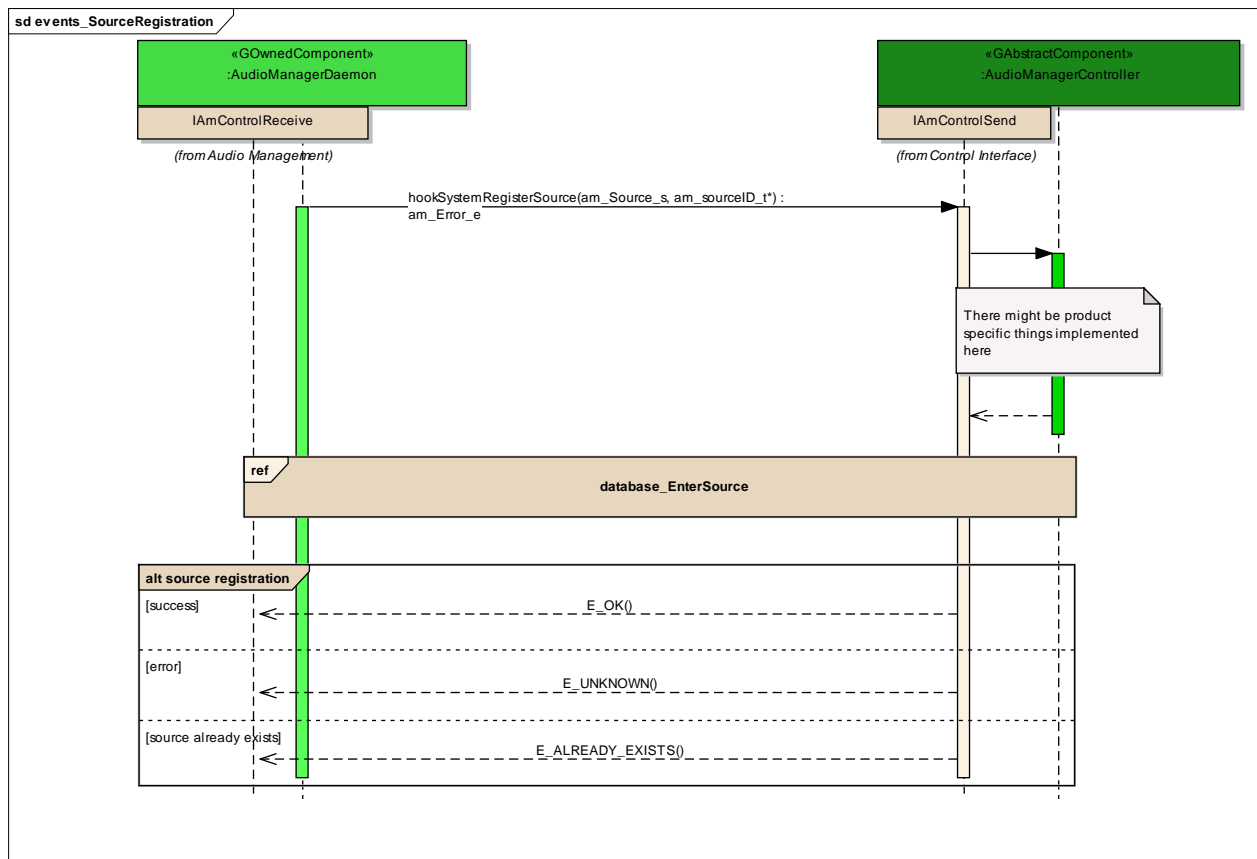
Figure: 98

database_EnterSource

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SourceRegistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{DF50F026-A92D-4f4f-8B05-571F2AA9270F}

source registration

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SourceRegistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{AFB5E6B9-1B90-4631-98F8-EC06E894C884}

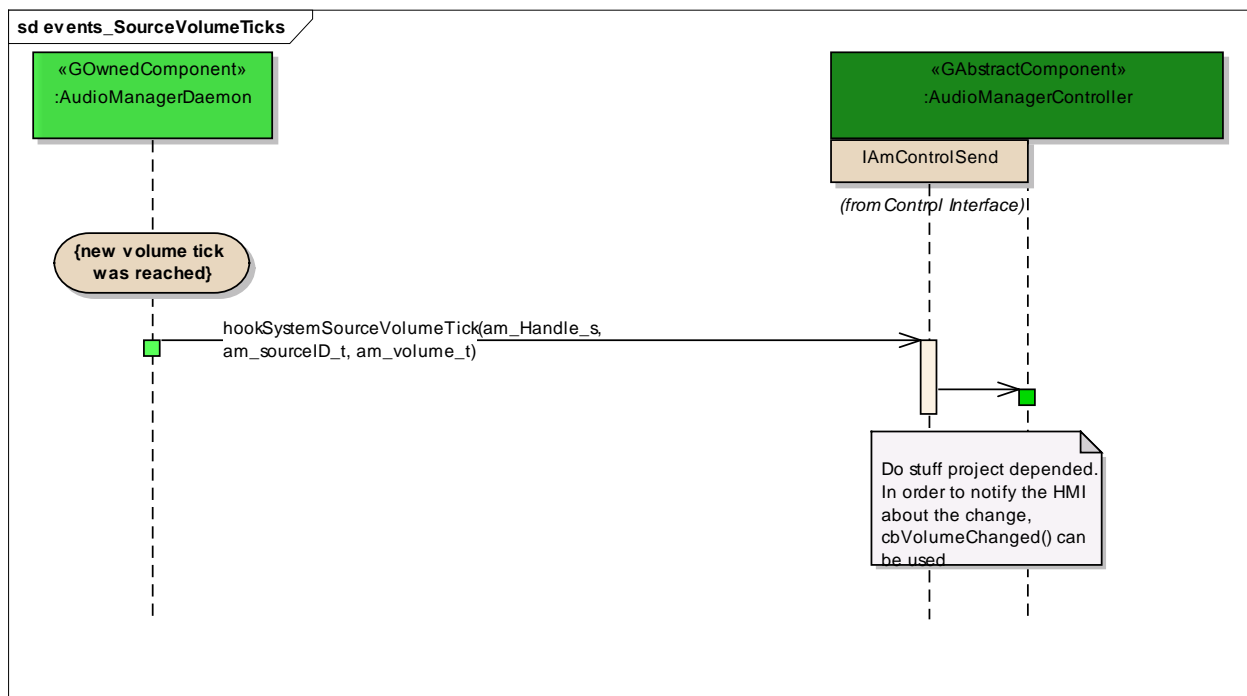# events_SourceDeregistration

Type: **Package**
Status: Reworked. Version 1.0. Phase 1.0.
Package: System Events
Detail: *Created on 13.09.2011. Last modified on 11.10.2011*
GUID: {1B1D4155-C360-41d9-81B4-F261B100B9FA}

events_SourceDeregistration - *(Interaction diagram)*

Created By: christian *on* 13.09.2011
Last Modified: 11.10.2011
Version: 1.0. *Locked:* False
GUID: {910ADD16-9685-4e68-8681-B6117EBE95EE}
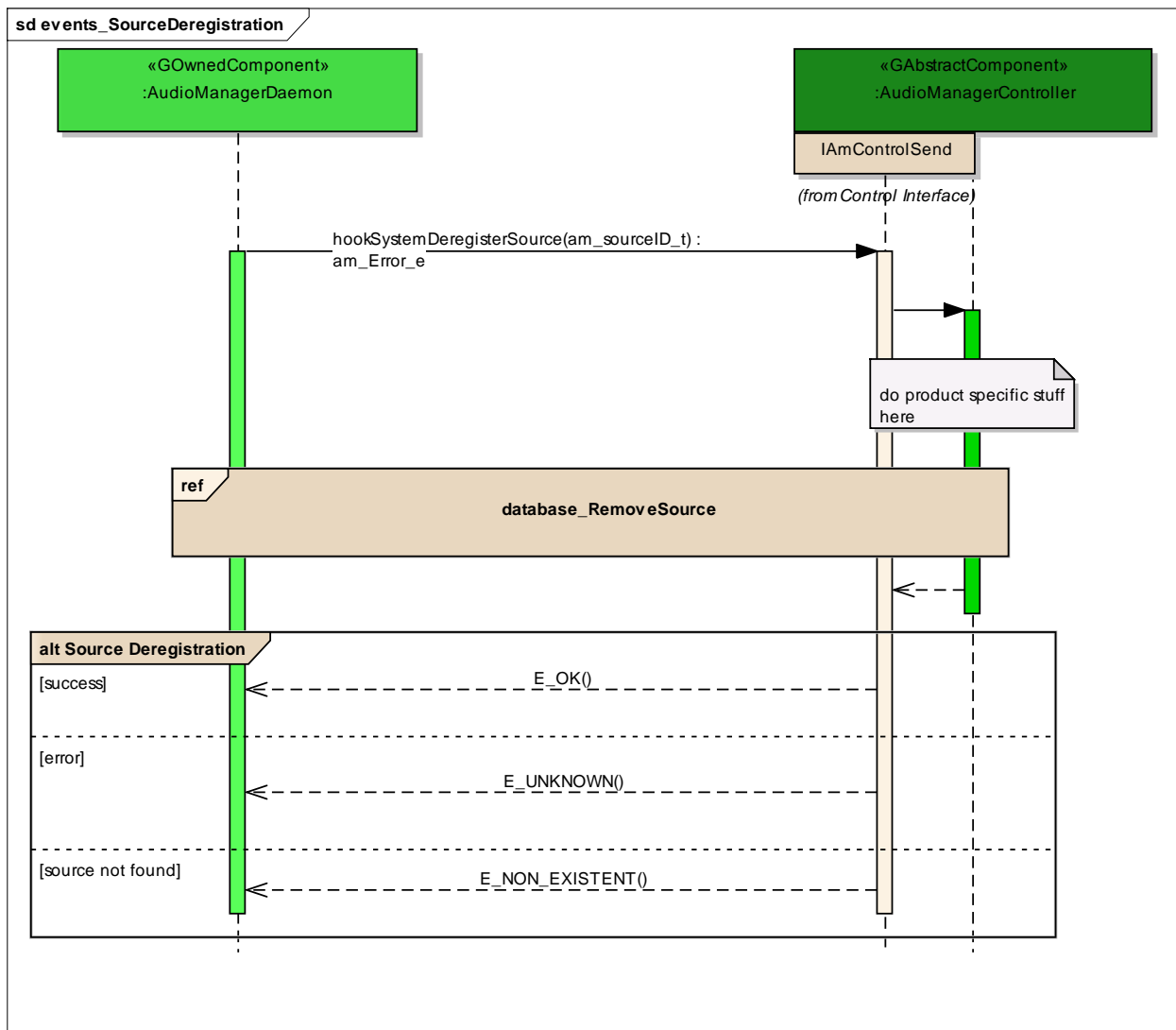
Figure: 99

Source Deregistration

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SourceDeregistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{937D4582-BD4A-436e-82F1-509667A18845}

database_RemoveSource

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SourceDeregistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{27284530-0880-4d92-BE6C-14B7E8D40D8F}

events_DomainRegistration

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 19.08.2011. Last modified on 11.10.2011*
GUID:	{863203EB-A5D5-4c09-9EEC-AF7ABE44648F}

events_DomainRegistration - *(Interaction diagram)*

Created By:	christian *on* 19.08.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{C89198FC-734B-4429-87BA-3D082BD5C396}
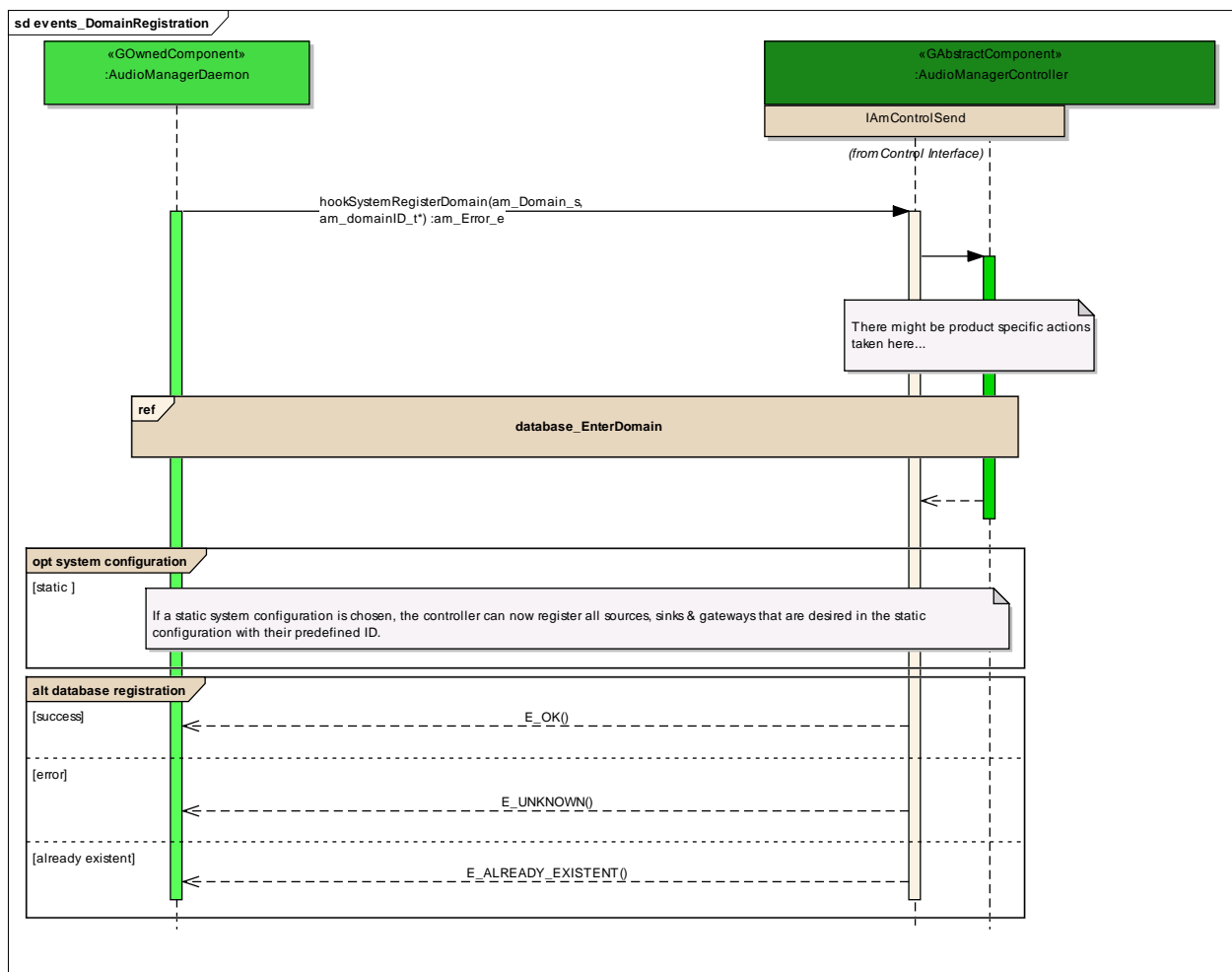
Figure: 100

database registration

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainRegistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{02E478A3-23B4-40f5-AD71-8E6133ABF408}

database_EnterDomain

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.

product specific

Type: **InteractionFragment**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_DomainRegistration   *Keywords:*
Detail: *Created on 19.08.2011. Last modified on 19.08.2011.*
GUID: {094370A2-BEC4-4248-8079-4809DAD6598F}

system configuration

Type: **InteractionFragment**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_DomainRegistration   *Keywords:*
Detail: *Created on 10.10.2011. Last modified on 10.10.2011.*
GUID: {B24C7B5B-3B81-4dae-821C-B8564F0CD4B3}

events_DomainDeregistration

Type: **Package**
Status: Reworked. Version 1.0. Phase 1.0.
Package: System Events
Detail: *Created on 13.09.2011. Last modified on 11.10.2011*
GUID: {AE4E8C93-05A2-413b-B956-28A45135776F}

events_DomainDeregistration - *(Interaction diagram)*

Created By: christian *on* 13.09.2011
Last Modified: 13.10.2011
Version: 1.0. *Locked:* False

GUID: {A1E82F6A-1189-4e5f-A3E3-1ECD803072DC}

sd events_DomainDeregistration

«GOwnedComponent»
:AudioManagerDaemon

the order of the
removal of gateways,
sinks and sources is
critical!

«GAbstractComponent»
:AudioManagerController

IAmControlSend

(from Control Interface)

hookSystemDeregisterDomain(am_domainID_t)
:am_Error_e

Do product specific stuff here

ref database_GetListGatewaysOfDomain

loop gateways
[for all gateways]

ref database_RemoveGateway

ref database_GetListCrossfadersOfDomain

loop crossfaders

ref database_RemoveCrossfader

ref database_GetListSinksOfDomain

loop sinks
[for all sinks that do not belong to a gateway]

ref database_RemoveSink

ref database_GetListSourcesOfDomain

loop sources
[for all sources that do not belong to gateways]

ref database_RemoveSource

ref database_RemoveDomain

alt removing of domain
[success]

[error]
E_OK()

[domain not found]
E_UNKNOWN()

E_NON_EXISTENT()

Figure: 101

crossfaders

Type: **InteractionFragment**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_DomainDeregistration   *Keywords:*
Detail: *Created on 13.10.2011. Last modified on 13.10.2011.*
GUID: {22446648-F875-413f-A914-7793559EEFCC}

database_GetListCrossfaders

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_DomainDeregistration   *Keywords:*
Detail: *Created on 13.10.2011. Last modified on 13.10.2011.*
GUID: {94E46FF3-E61C-4512-8AD0-1EAFEF676DBE}

database_GetListCrossfadersOfDomain

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_DomainDeregistration   *Keywords:*
Detail: *Created on 13.10.2011. Last modified on 13.10.2011.*
GUID: {AE7354A4-9F13-4dab-84D4-D423E77472C4}
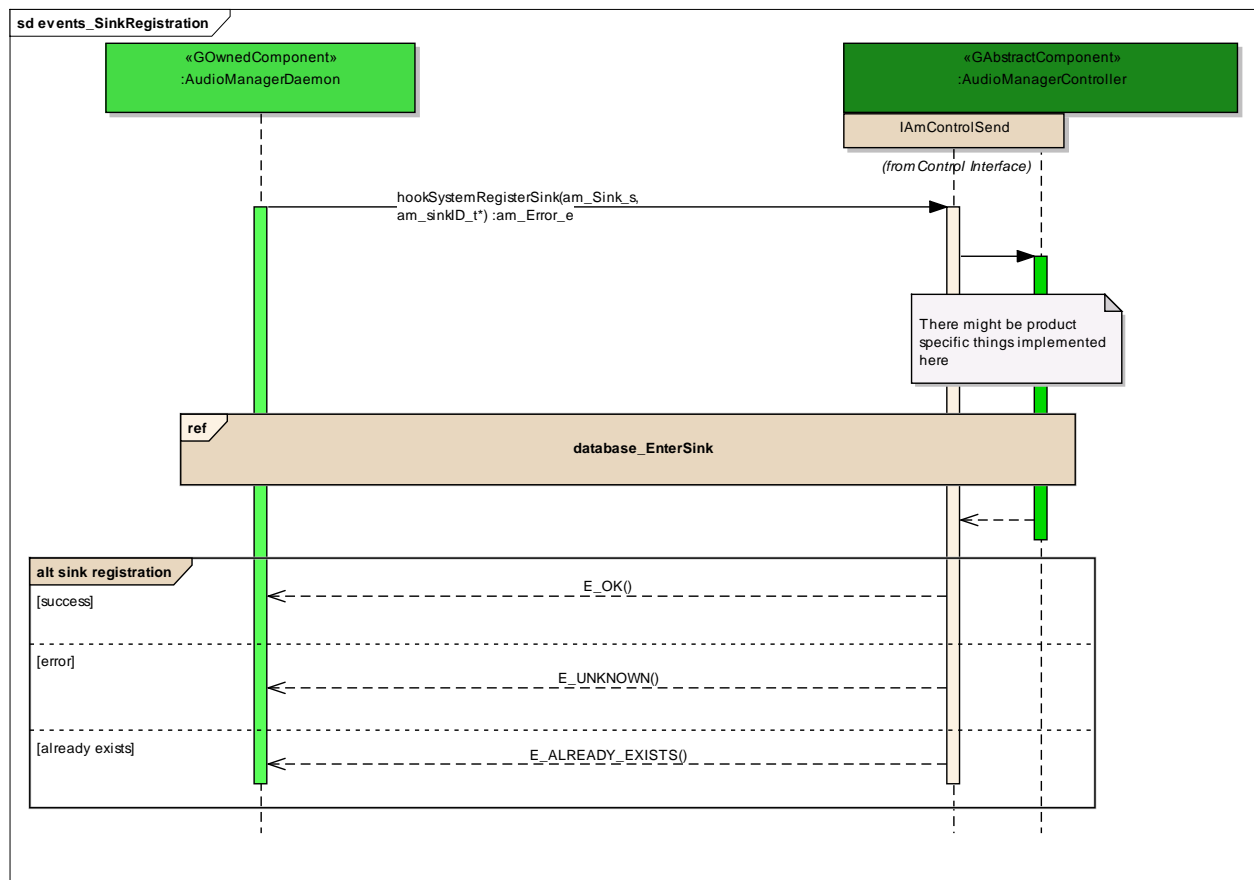
database_GetListGatewaysOfDomain

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_DomainDeregistration   *Keywords:*
Detail: *Created on 10.10.2011. Last modified on 10.10.2011.*

| GUID: | {B4387318-5B86-47da-BAB1-E68E82505351} |

database_GetListSinksOfDomain

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainDeregistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{BFA8694C-B2CC-4472-A8F7-5DD8FD6E89EE}

database_GetListSourcesOfDomain

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainDeregistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{2CE3AFE6-235B-42a9-AF5A-B1C67CD63507}

database_RemoveCrossfader

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainDeregistration *Keywords:*
Detail:	*Created on 13.10.2011. Last modified on 13.10.2011.*
GUID:	{B1E9664B-2445-4c19-B126-F4F90B022B2C}

database_RemoveDomain

Type:	**InteractionOccurrence**

Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainDeregistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{6536E99B-2889-467d-8729-B03B59E4FE16}

database_RemoveGateway

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainDeregistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{74E1F413-5C60-4d0e-B930-E94D4281E661}

database_RemoveSink

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainDeregistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{8D29CA39-C757-4f6c-B391-678FCE916360}

database_RemoveSource

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainDeregistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{D59E9DE7-CD37-49af-8068-FCCFFEC25470}

gateways

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainDeregistration *Keywords:*
Detail:	*Created on 13.09.2011. Last modified on 13.09.2011.*
GUID:	{39BB4E55-06F4-4293-BF03-DB7A668F19AB}

removing of domain

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainDeregistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{F231EFC8-8F5A-44f6-A579-8DB493C1AE52}

sinks

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainDeregistration *Keywords:*
Detail:	*Created on 13.09.2011. Last modified on 13.09.2011.*
GUID:	{45FDC8C7-2714-4412-982D-04BE031AEE3D}

sources

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_DomainDeregistration *Keywords:*
Detail:	*Created on 13.09.2011. Last modified on 13.09.2011.*
GUID:	{A8B8F356-0FDC-43d4-9428-B4918BA8D3D6}

events_GatewayRegistration

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 22.08.2011. Last modified on 11.10.2011*
GUID:	{425BE348-A8A7-439e-B2BF-F2EAF6A78A9A}

events_GatewayRegistration - *(Interaction diagram)*

Created By:	christian *on* 22.08.2011
Last Modified:	11.10.2011
Version:	1.0. *Locked:* False
GUID:	{F99997B5-4258-4495-BCA3-F7F777752DE7}
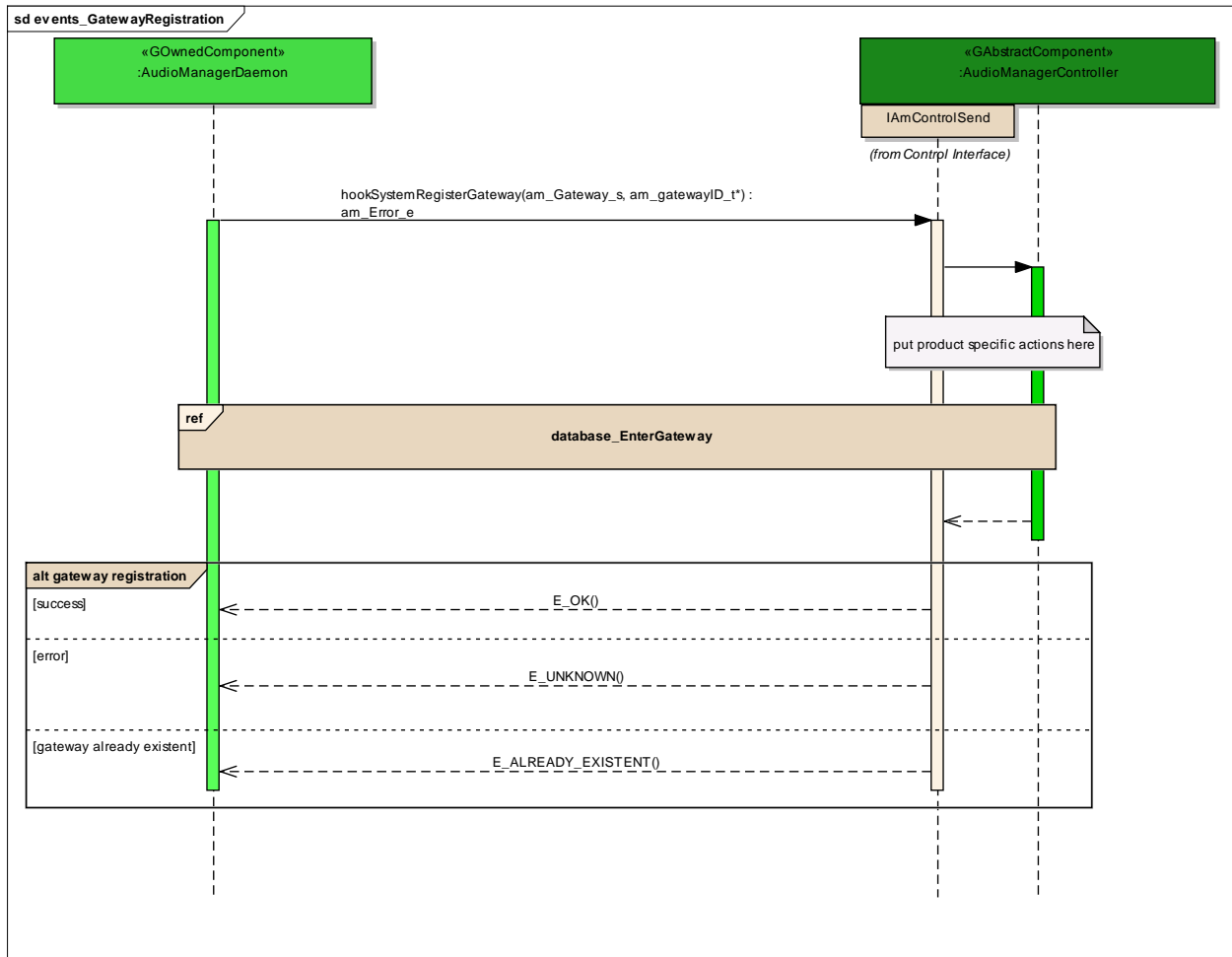
Figure: 102

database_EnterGateway

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_GatewayRegistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{DA013B16-4FD6-4912-8DFB-3A066DA72CC2}

gateway registration

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_GatewayRegistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{0FC196B8-74D1-4d69-BF4F-6E6BAC5FC32E}

events_GatewayDeregistration

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 13.09.2011. Last modified on 11.10.2011*
GUID:	{055C1585-6A43-49be-801F-A34DAEBBA7AD}

events_GatewayDeregistration - *(Interaction diagram)*

Created By:	christian *on* 13.09.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{52F4BCEB-7F02-4fc2-95CE-110519892523}
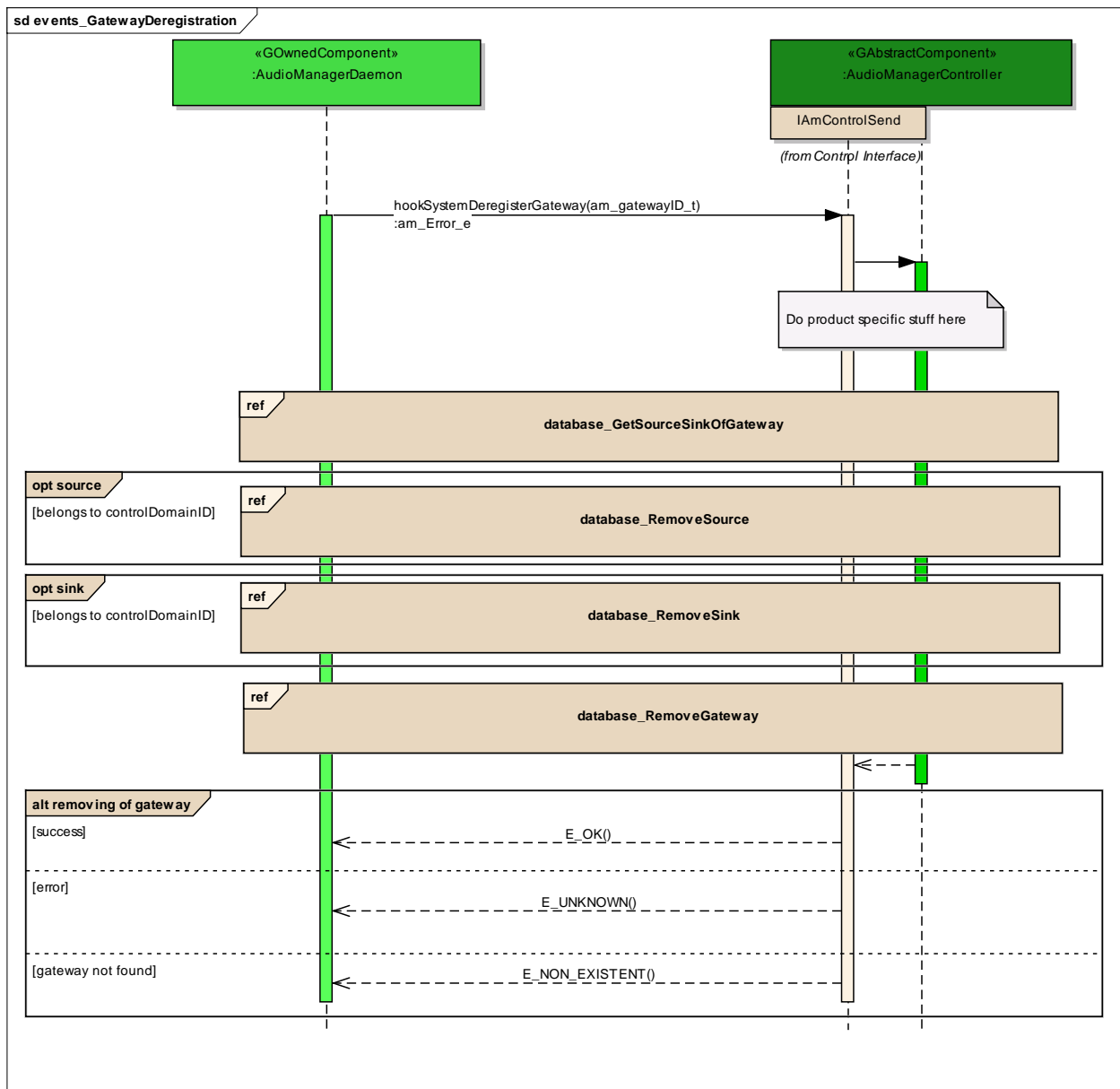
Figure: 103

database_GetSourceSinkOfGateway

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_GatewayDeregistration *Keywords:*
Detail:	*Created on 10.10.2011. Last modified on 10.10.2011.*
GUID:	{C98ABA96-2DA9-455a-8DB4-C0F7E5269638}

database_RemoveGateway

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_GatewayDeregistration   *Keywords:*
Detail: *Created on 10.10.2011. Last modified on 10.10.2011.*
GUID: {0D9C98AF-0573-434e-9EFB-D9BED986D501}

database_RemoveSink

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_GatewayDeregistration   *Keywords:*
Detail: *Created on 10.10.2011. Last modified on 10.10.2011.*
GUID: {632D8EB0-F825-4bb2-A773-D3FE41C57F00}

database_RemoveSource

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_GatewayDeregistration   *Keywords:*
Detail: *Created on 10.10.2011. Last modified on 10.10.2011.*
GUID: {66ECA4E4-F9BD-4c5b-B906-B60002334400}

removing of gateway

Type: **InteractionFragment**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_GatewayDeregistration   *Keywords:*
Detail: *Created on 10.10.2011. Last modified on 10.10.2011.*
GUID: {72DE5127-A4CB-4bb9-A2E9-83D06263F597}

sink

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_GatewayDeregistration *Keywords:*
Detail:	*Created on 11.10.2011. Last modified on 11.10.2011.*
GUID:	{07D97FA1-79F7-42e1-AA71-91A43A6EBF8A}

source

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_GatewayDeregistration *Keywords:*
Detail:	*Created on 11.10.2011. Last modified on 11.10.2011.*
GUID:	{BF574A3D-9604-44c5-8B23-DD5EDA34A1E1}

events_CrossfaderRegistration

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011*
GUID:	{AA4FE49D-C60D-4d3a-B403-912D7C1D7A92}

events_CrossfaderRegistration - *(Interaction diagram)*

Created By:	christian *on* 12.10.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{84B47D16-4CE4-4f69-B003-293CECB52728}
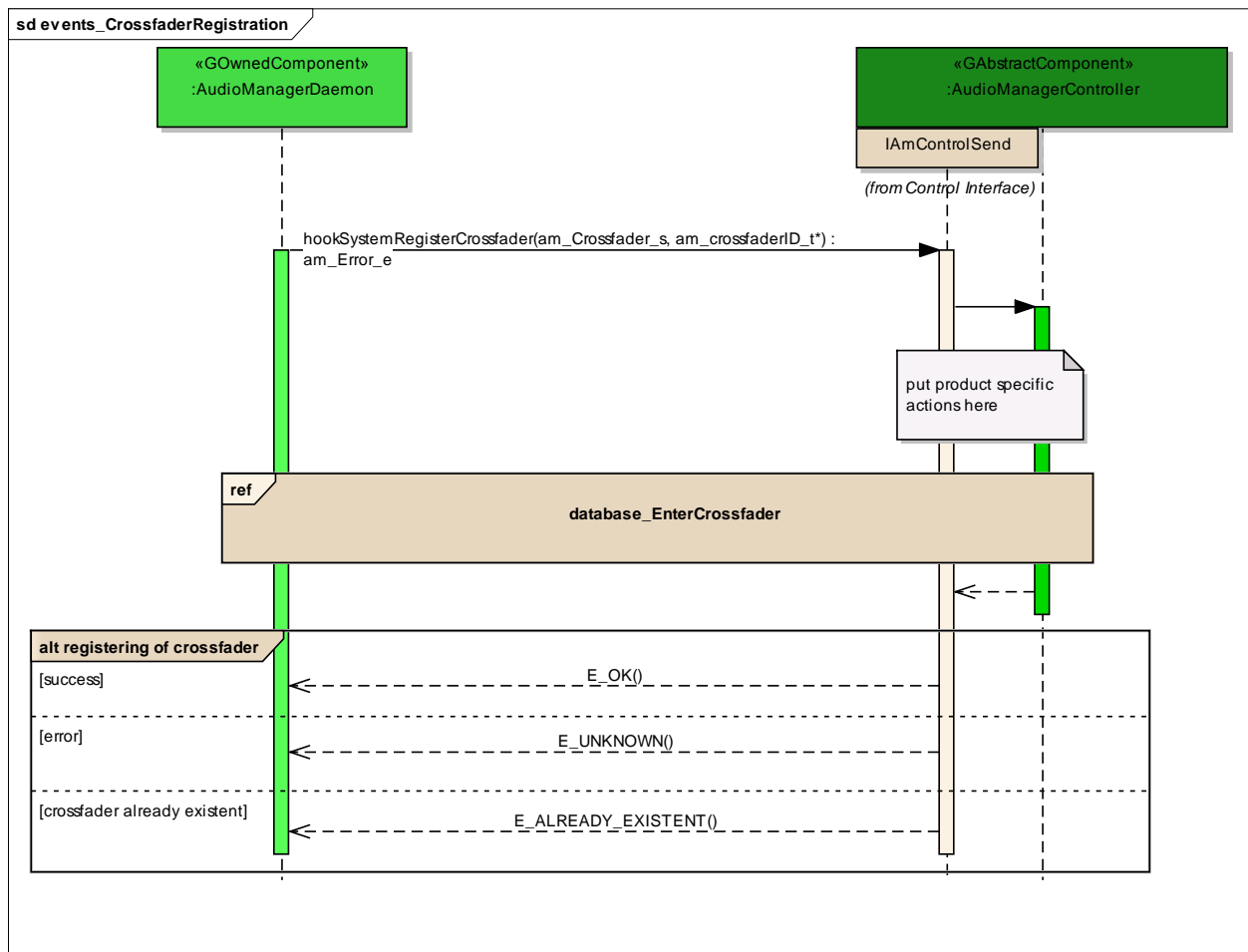
Figure: 104

database_EnterCrossfader

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_CrossfaderRegistration *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{341DF38E-9255-43b8-B22B-36EEAFF983CB}

registering of crossfader

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_CrossfaderRegistration *Keywords:*

events_CrossfaderDeregistration

Type: **Package**
Status: Proposed. Version 1.0. Phase 1.0.
Package: System Events
Detail: *Created on 12.10.2011. Last modified on 12.10.2011*
GUID: {7443B725-1C3A-4094-9EE3-A09875D2BA2D}

events_CrossfaderDeregistration - *(Interaction diagram)*
Created By: christian *on* 12.10.2011
Last Modified: 12.10.2011
Version: 1.0. *Locked:* False
GUID: {C1263C19-9D59-42d2-B63D-1DA82A8D036A}
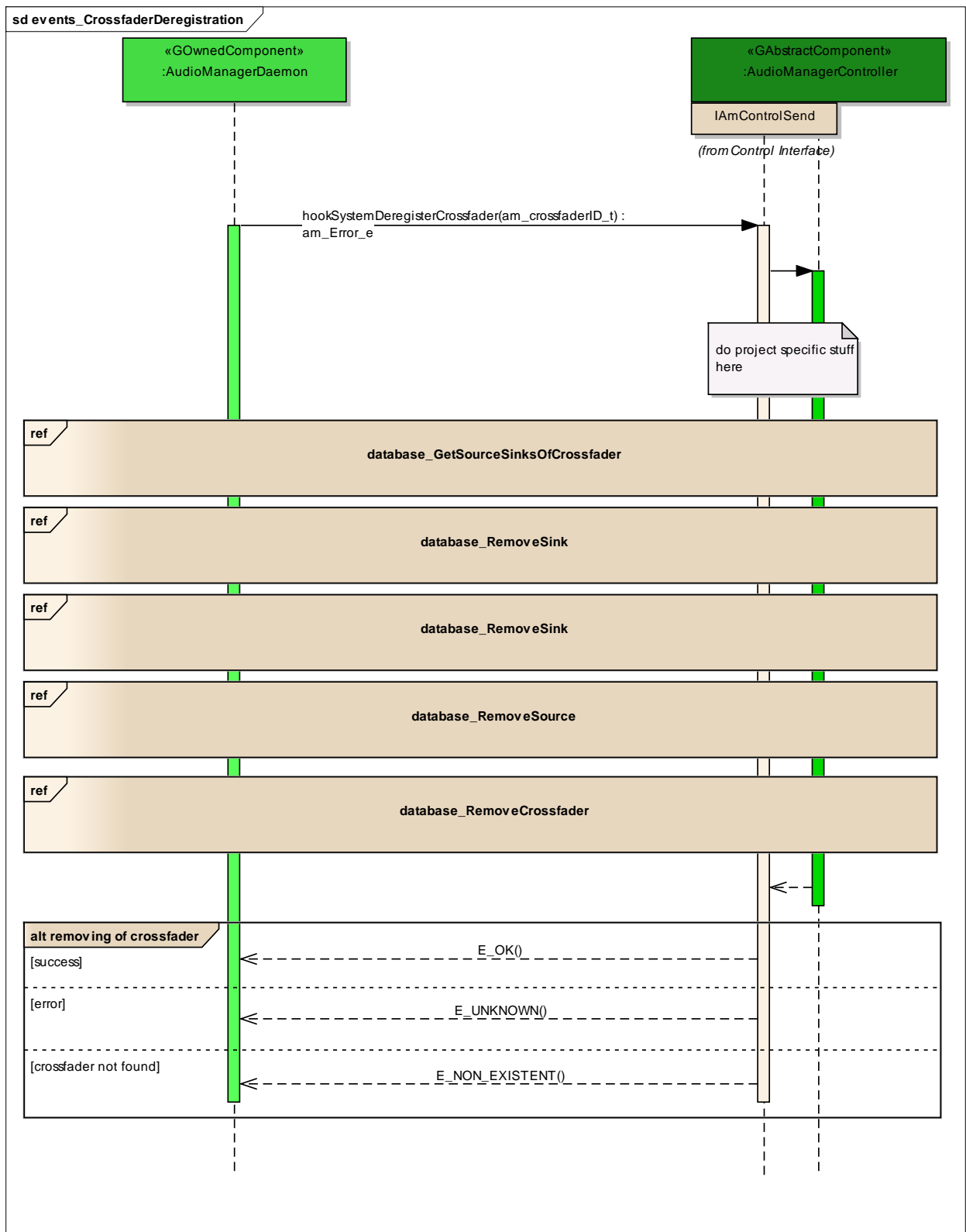
Figure: 105

database_GetSourceSinksOfCrossfader

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_CrossfaderDeregistration *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{BC0FE970-B43B-4aff-9EE4-EB01F9B9EF24}

database_RemoveCrossfader

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_CrossfaderDeregistration *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{AAAC861B-297C-47a2-A990-4196C9AAABCE}

database_RemoveSink

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_CrossfaderDeregistration *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{B3285ECF-6C1D-48d7-B2CD-4B7D78C203F2}

database_RemoveSink

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_CrossfaderDeregistration *Keywords:*
Detail:	*Created on 12.10.2011. Last modified on 12.10.2011.*
GUID:	{44EE5AAA-9A20-419d-94CC-70328BA09263}

database_RemoveSource

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_CrossfaderDeregistration   *Keywords:*
Detail: Created on 12.10.2011. Last modified on 12.10.2011.
GUID: {05A39DC5-6EB4-41d0-83CB-F1CF2D73998E}

removing of crossfader

Type: **InteractionFragment**
Status: Proposed. Version 1.0. Phase 1.0.
Package: events_CrossfaderDeregistration   *Keywords:*
Detail: Created on 12.10.2011. Last modified on 12.10.2011.
GUID: {E0CFF33A-0C41-4be5-88F0-5C5864E88BB8}

events_TimingInformationChanged

Type: **Package**
Status: Reworked. Version 1.0. Phase 1.0.
Package: System Events
Detail: Created on 23.09.2011. Last modified on 11.10.2011
GUID: {5EBFD294-B464-434b-9822-6D9A338FC168}

events_TimingInformationChanged - *(Interaction diagram)*
Created By: christian *on* 23.09.2011
Last Modified: 12.10.2011
Version: 1.0. *Locked:* False
GUID: {18C60FF9-809B-41be-834F-FDB8ED263B1F}
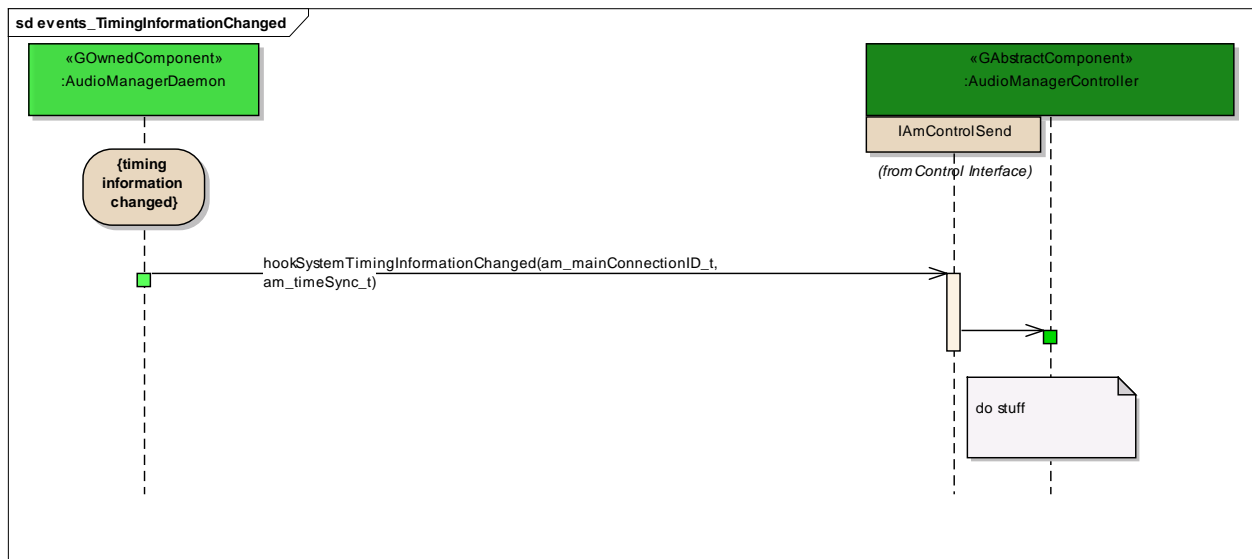
Figure: 106

events_ReceiveChangedData

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 21.09.2011. Last modified on 11.10.2011*
GUID:	{07256723-058F-4296-BEBD-11DC7BB5084C}

events_ReceiveChangedData - *(Interaction diagram)*

Created By:	christian *on* 21.09.2011
Last Modified:	12.10.2011
Version:	1.0. *Locked:* False
GUID:	{2233B6E8-251D-4978-A00C-04850BABC7D5}
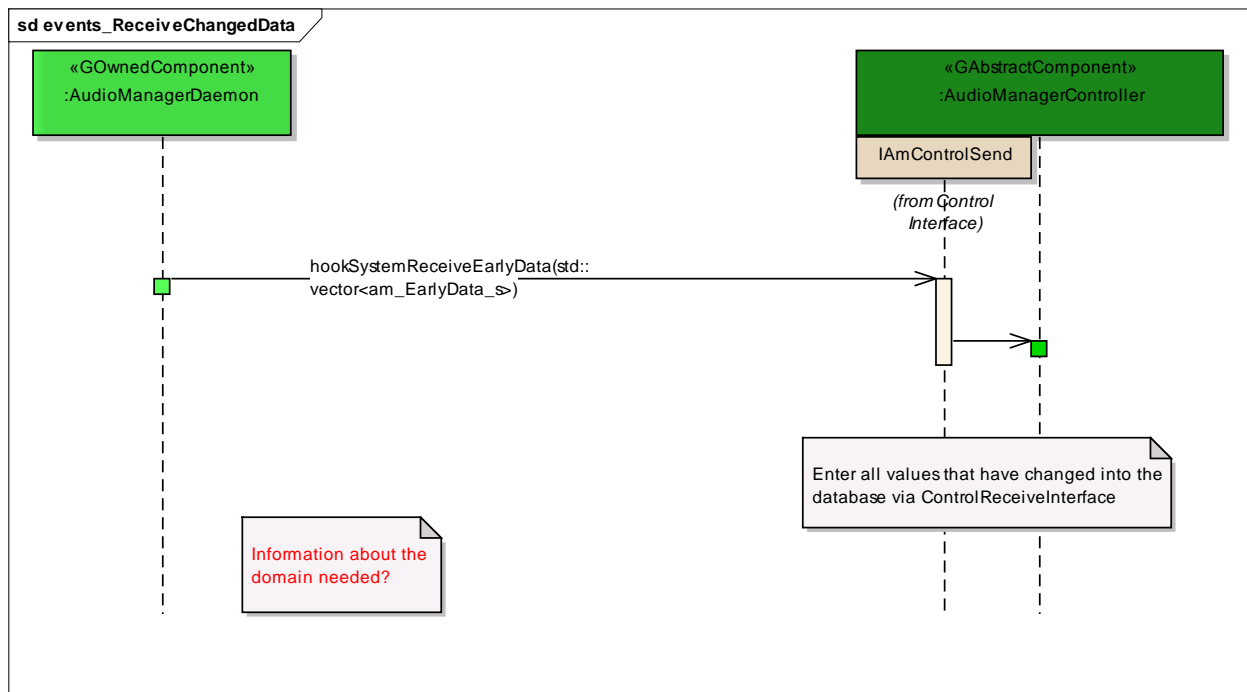
Figure: 107

product specific

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_ReceiveChangedData *Keywords:*
Detail:	*Created on 21.09.2011. Last modified on 21.09.2011.*
GUID:	{FEFCE9C7-E3A0-48e1-86E6-EA46F3E4B10A}

events_LowLevelInterruptStateChange

Type:	**Package**
Status:	Reworked. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 09.09.2011. Last modified on 11.10.2011*
GUID:	{CDB303B6-DBFF-4c52-8979-1A93E5F4CB40}

events_LowLevelInterruptStateChange - *(Interaction diagram)*
Created By: christian *on* 09.09.2011
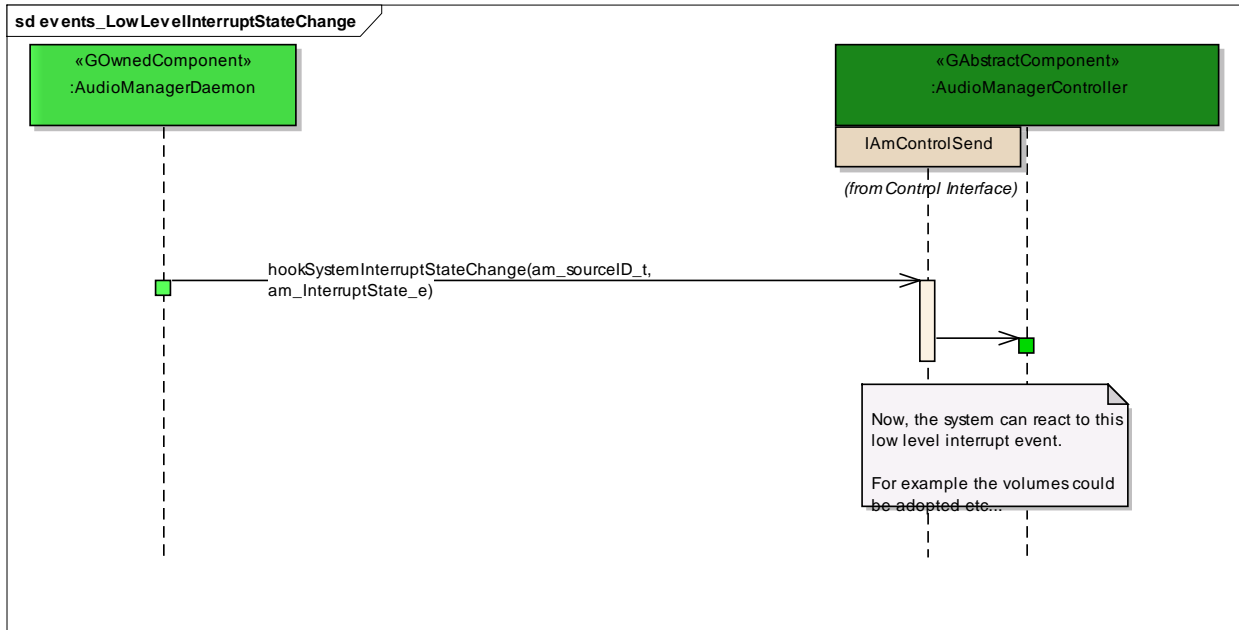
Figure: 108

events_SourceDataUpdate

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	Created on 28.06.2012. Last modified on 28.06.2012
GUID:	{F998D4F9-918A-4472-81E1-C370318B176E}

events_SourceDataUpdate - *(Interaction diagram)*
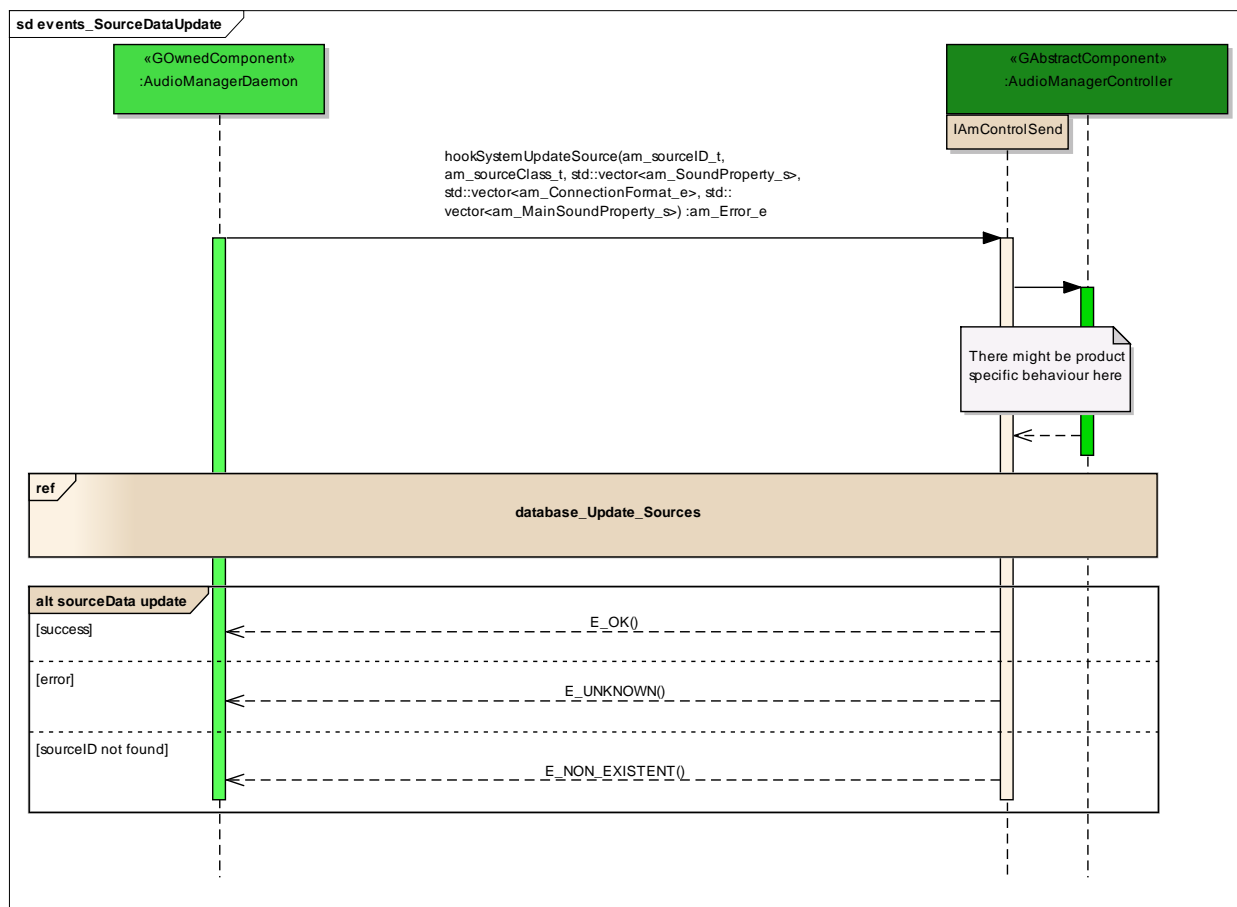
Figure: 109

database_Update_Sources

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SourceDataUpdate *Keywords:*
Detail:	Created on 28.06.2012. Last modified on 28.06.2012.
GUID:	{85582B39-497F-4f19-99CA-0CDE3B07F128}

sourceData update

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SourceDataUpdate *Keywords:*
Detail:	Created on 28.06.2012. Last modified on 28.06.2012.

GUID: {8B6A12F8-3392-4c88-9F4F-3B5FDFBA37C3}

events_GatewayDataUpdate

Type: **Package**
Status: Proposed. Version 1.0. Phase 1.0.
Package: System Events
Detail: *Created on 29.06.2012. Last modified on 29.06.2012*
GUID: {27A82473-DEC3-4fff-8B04-93BD7283EF2B}

events_GatewayDataUpdate - *(Interaction diagram)*

Created By: q232968 *on* 29.06.2012
Last Modified: 28.11.2012
Version: 1.0. *Locked:* False
GUID: {D9C2509E-AA1A-44ab-AFD8-1A20A9F5F31A}
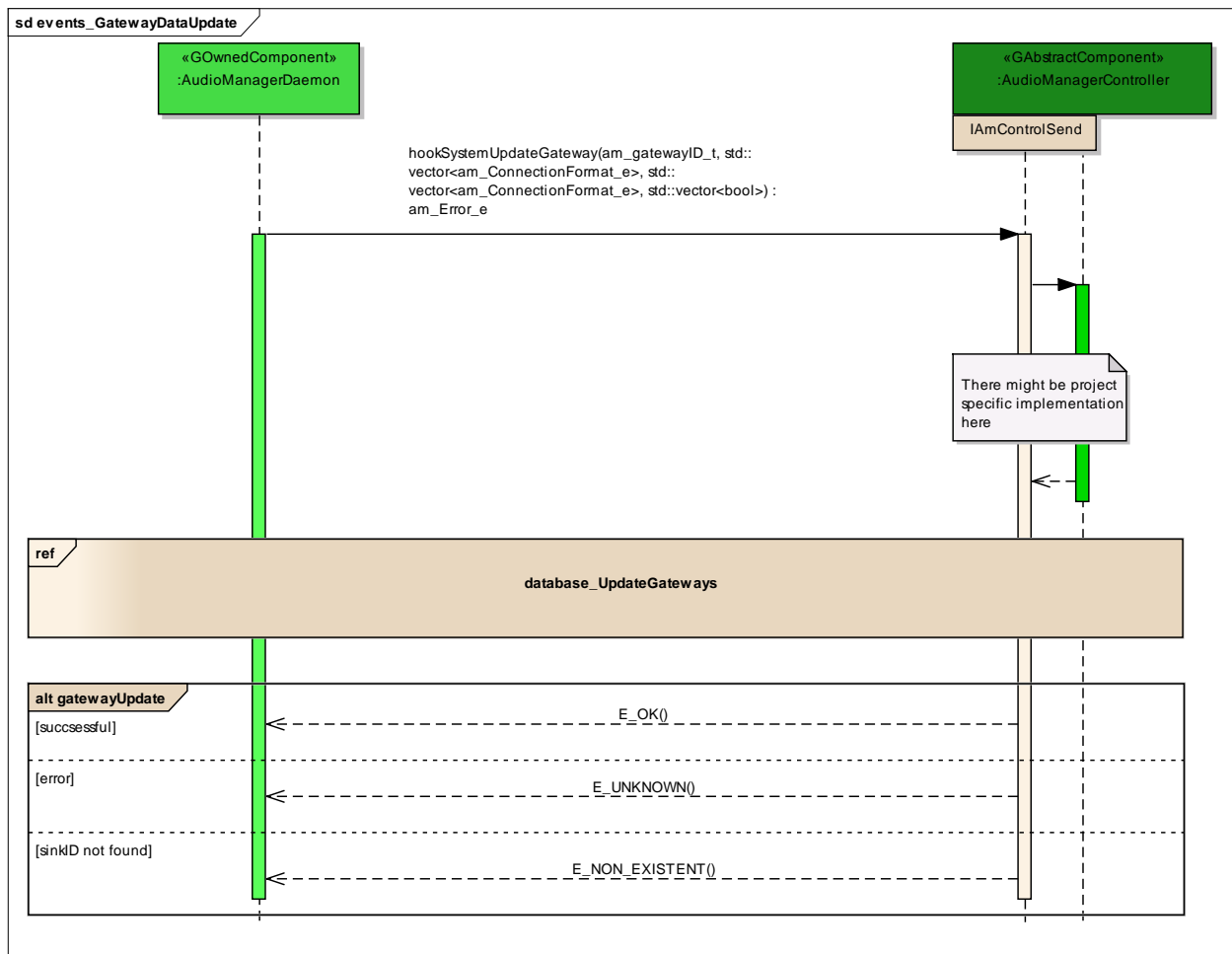
Figure: 110

database_UpdateGateways

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_GatewayDataUpdate *Keywords:*
Detail:	*Created on 29.06.2012. Last modified on 29.06.2012.*
GUID:	{66A40AC1-583D-4962-860B-2678F4E79856}

database_UpdateSinks

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_GatewayDataUpdate *Keywords:*

gatewayUpdate

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_GatewayDataUpdate *Keywords:*
Detail:	*Created on 29.06.2012. Last modified on 29.06.2012.*
GUID:	{8DF9227F-72AE-489a-93C5-7BB325FB5EC7}

events_SinkDataUpdate

Type:	**Package**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	System Events
Detail:	*Created on 29.06.2012. Last modified on 29.06.2012*
GUID:	{6D62ABB0-560F-48b9-8917-3D601CF05D2A}

events_SinkDataUpdate - *(Interaction diagram)*

Created By:	q232968 *on* 29.06.2012
Last Modified:	28.11.2012
Version:	1.0. *Locked:* False
GUID:	{9BDB953A-8042-4a73-AC5D-26C26BA0F5E5}
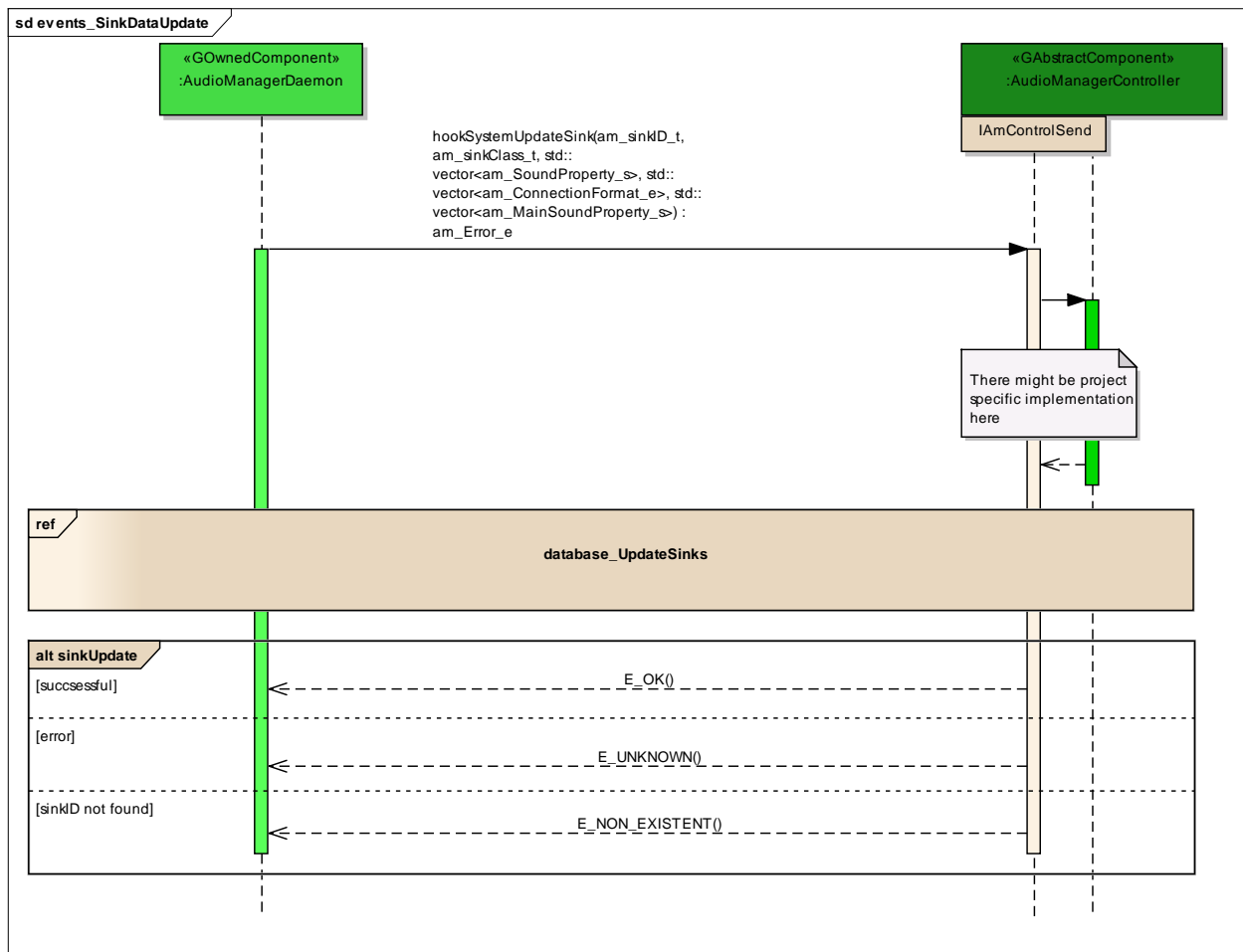
Figure: 111

database_UpdateSinks

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SinkDataUpdate *Keywords:*
Detail:	*Created on 29.06.2012. Last modified on 29.06.2012.*
GUID:	{7F1C314B-6689-4841-9B3A-CD89371B81C5}

sinkUpdate

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	events_SinkDataUpdate *Keywords:*

Detail: *Created on 29.06.2012. Last modified on 29.06.2012.*
GUID: {1AB25CF6-626A-414f-9F12-D7DE120902AD}

9.5 NodeStateManagement

Type: **Package**
Status: Proposed. Version 1.0. Phase 1.0.
Package: Control Interface
Detail: *Created on 28.01.2013. Last modified on 28.01.2013*
GUID: {54788D12-E1AC-44ab-BC4F-2D8B1B5E3400}
These sequences are optional. The NSM support can be switched off !

NodeStateManagementRelations - *(Class diagram)*

Created By: q232968 *on* 31.01.2013
Last Modified: 31.01.2013
Version: 1.0. *Locked:* False
GUID: {F7E39BD4-DD4C-4fde-8DB5-A945BCA63E0B}
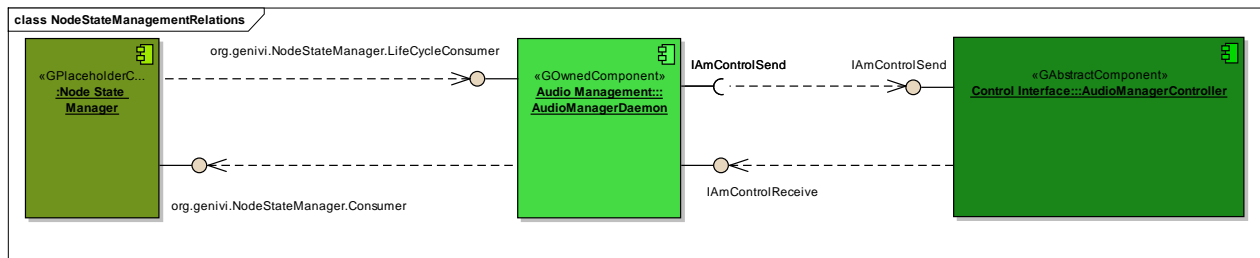
Figure: 112

GetInterfaceVersion - *(Interaction diagram)*

Created By: q232968 *on* 28.01.2013
Last Modified: 28.01.2013
Version: 1.0. *Locked:* False
GUID: {35C286C9-7838-483f-8B92-2B6199ABBBD9}
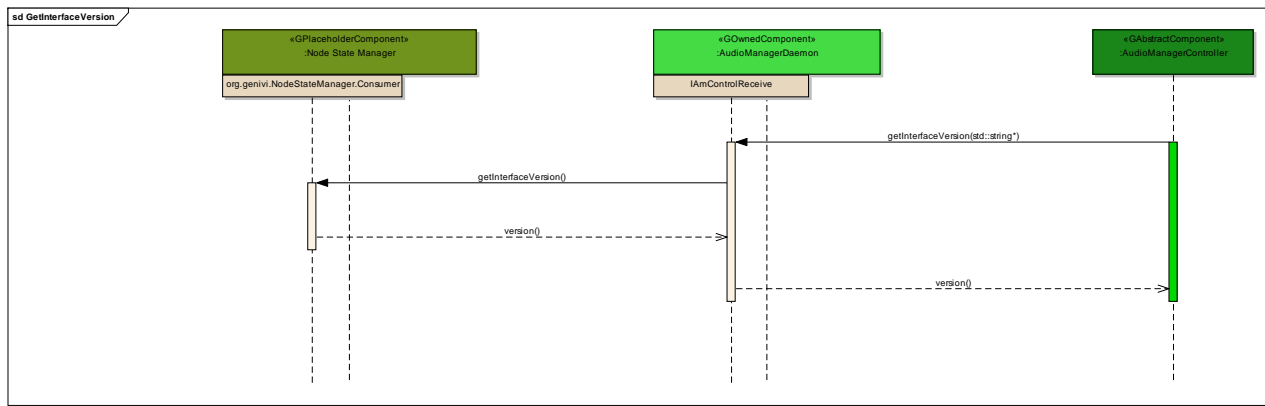
Figure: 113

GetNodeState - *(Interaction diagram)*

Created By: q232968 *on* 28.01.2013
Last Modified: 28.01.2013
Version: 1.0. *Locked:* False
GUID: {2DC27AD8-52CE-455e-BD1E-FCAB37861BD9}
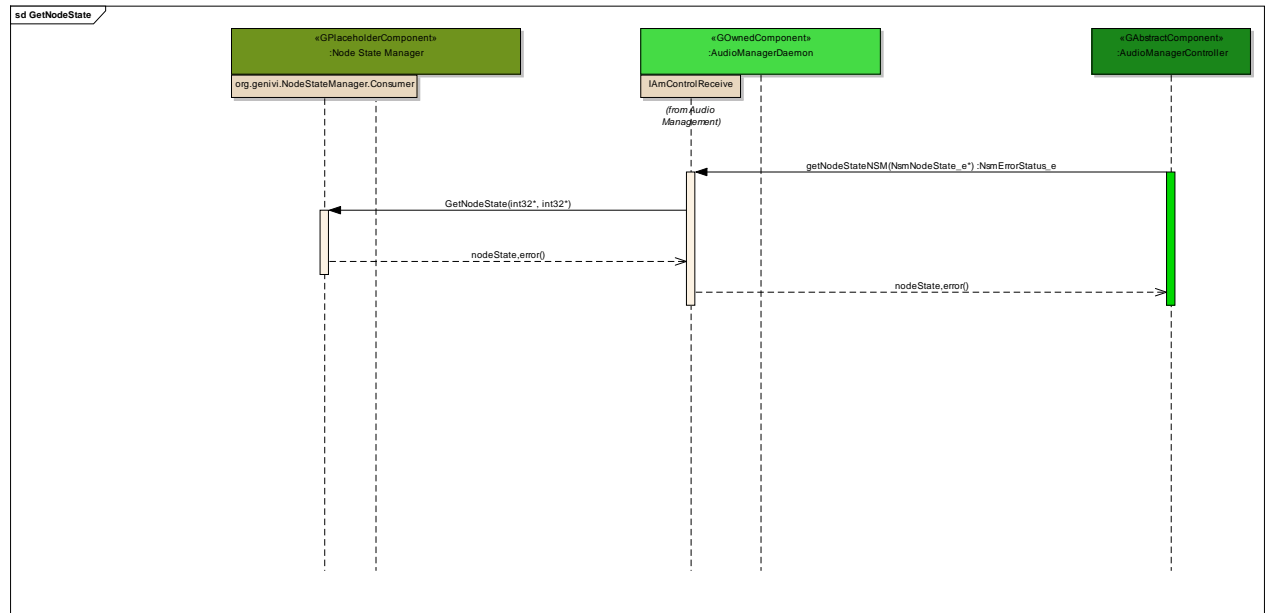
Figure: 114

GetRestartReason - *(Interaction diagram)*

Created By: q232968 *on* 28.01.2013
Last Modified: 28.01.2013
Version: 1.0. *Locked:* False
GUID: {1F4A4DEC-C2B7-456e-94E7-0AA2CAF99252}
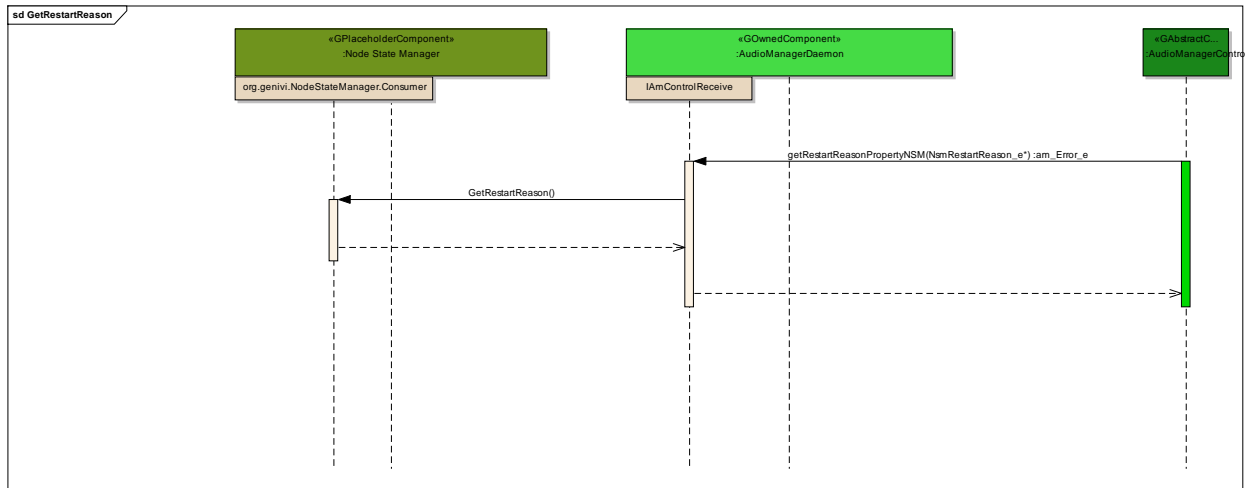
Figure: 115

GetRunningReson - *(Interaction diagram)*

Created By: q232968 *on* 28.01.2013
Last Modified: 28.01.2013
Version: 1.0. *Locked:* False
GUID: {8B220612-7981-462d-A3CD-BA0C1197956A}
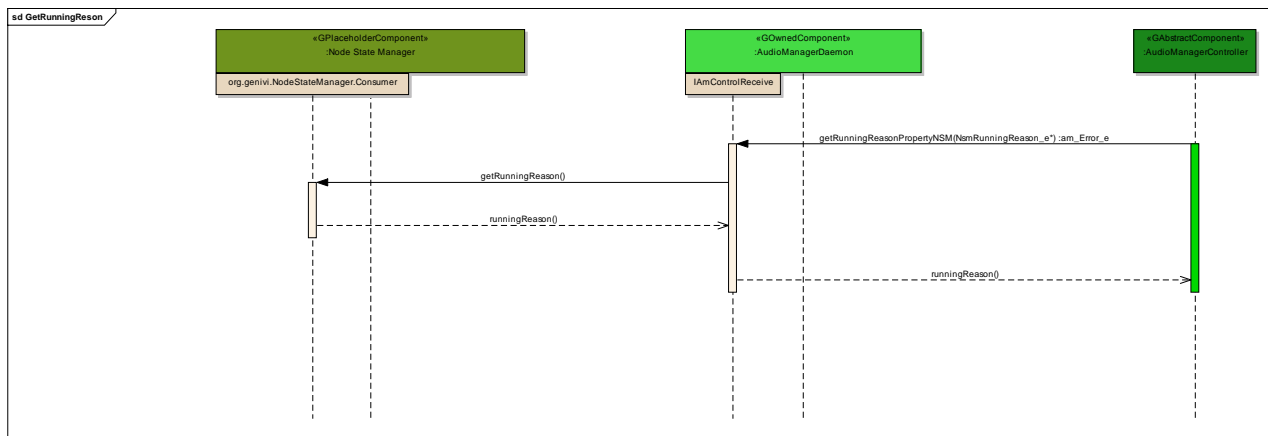
Figure: 116

GetSessionState - *(Interaction diagram)*

Created By: q232968 *on* 28.01.2013
Last Modified: 28.01.2013
Version: 1.0. *Locked:* False
GUID: {5D82D567-3F1D-4e78-A32A-9AE7C3ACD1E5}
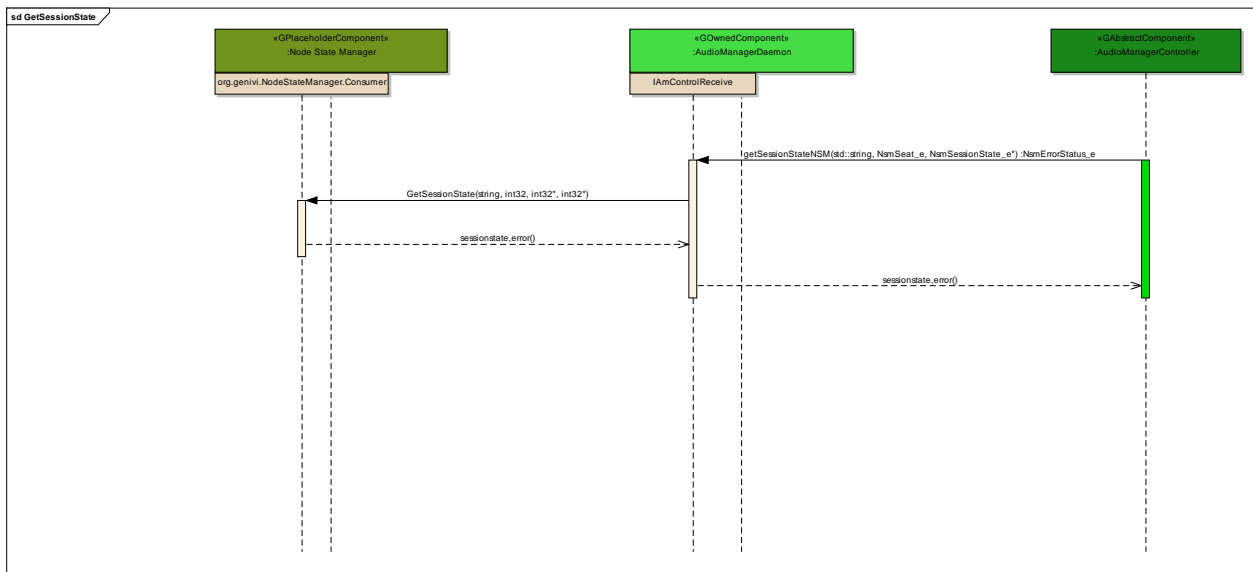
Figure: 117

GetShutdownReason - *(Interaction diagram)*

Created By:	q232968 *on* 28.01.2013
Last Modified:	28.01.2013
Version:	1.0. *Locked:* False
GUID:	{E0BDF9A7-304F-4852-B050-E128E2C4264F}
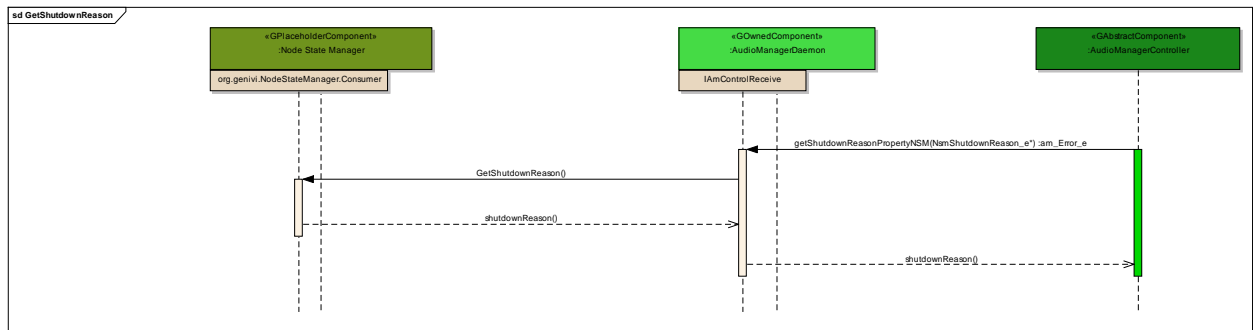
Figure: 118

Lifecylerequest - *(Interaction diagram)*

Created By:	q232968 *on* 28.01.2013
Last Modified:	31.01.2013
Version:	1.0. *Locked:* False
GUID:	{AA535FCF-5718-4b5d-B448-52A5D40BE9FE}
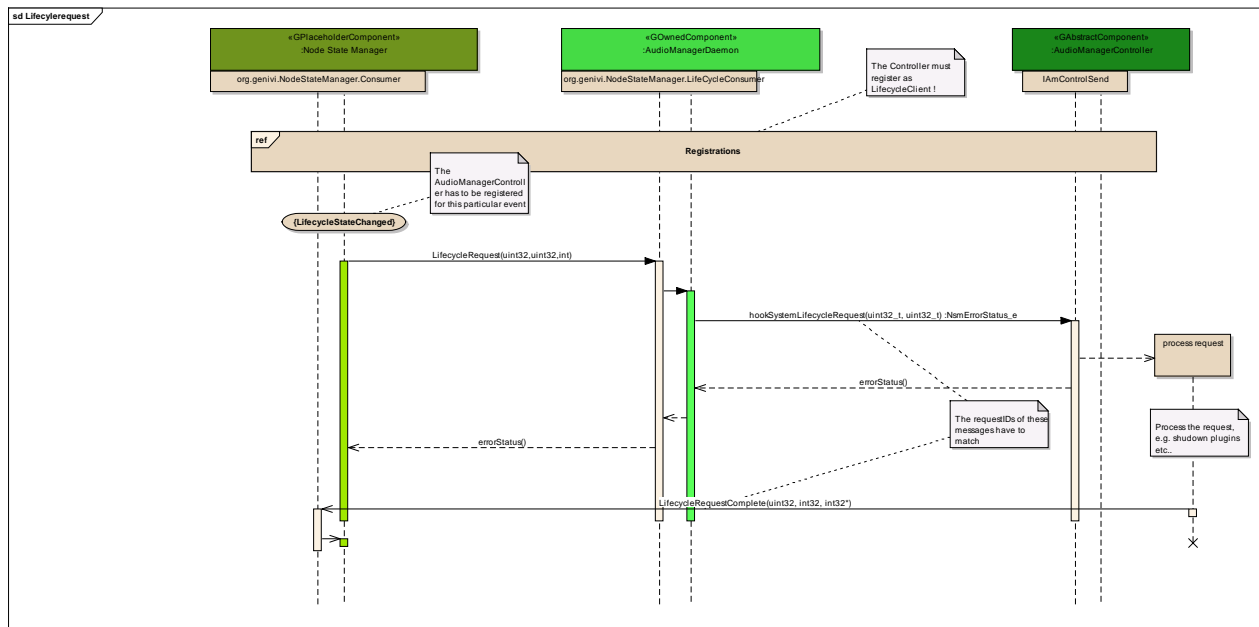
Figure: 119

getApplicationMode - *(Interaction diagram)*

Created By:	q232968 *on* 28.01.2013
Last Modified:	28.01.2013
Version:	1.0. *Locked:* False
GUID:	{85ABD946-C683-4f95-8D06-2DBB520D8FAE}
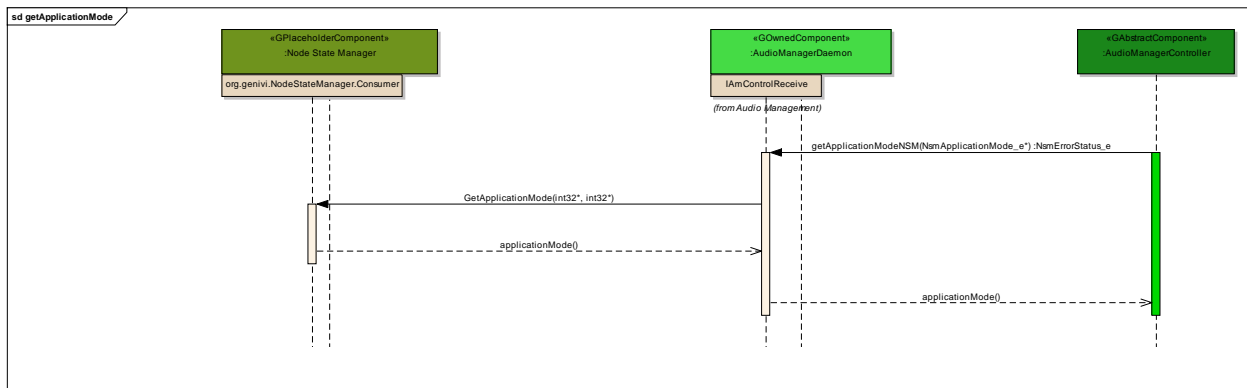
Figure: 120

unregisterLifecycleConsumer - *(Interaction diagram)*

Created By:	q232968 *on* 28.01.2013
Last Modified:	28.01.2013
Version:	1.0. *Locked:* False
GUID:	{902A1548-3F05-482c-A056-2DFD1DE95339}
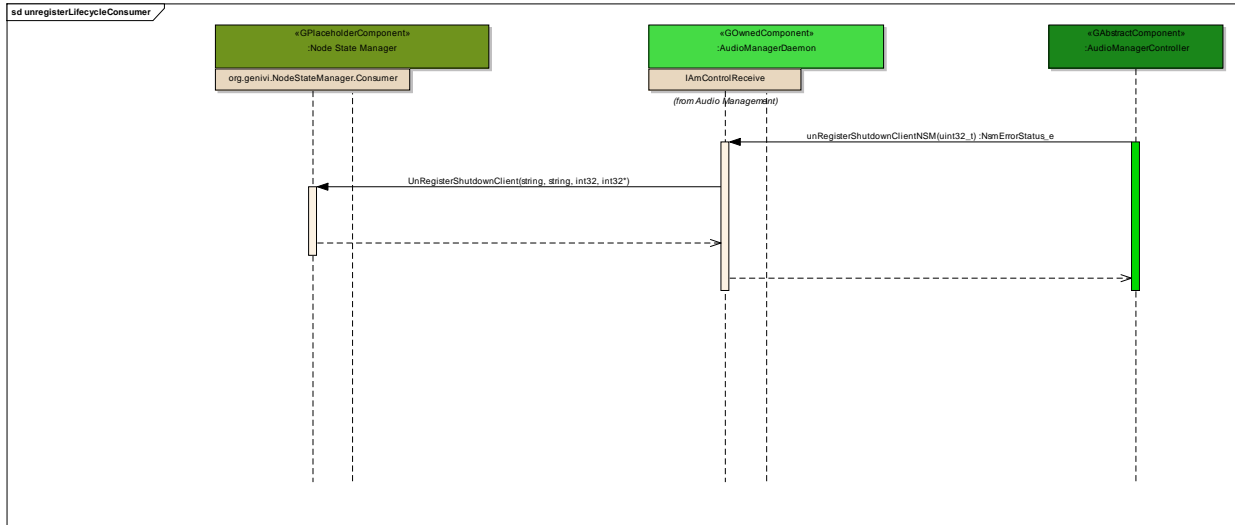
Figure: 121

Registrations - *(Interaction diagram)*

Created By:	q232968 *on* 28.01.2013
Last Modified:	28.01.2013
Version:	1.0. *Locked:* False
GUID:	{11351D31-D136-4ea4-9FC8-167B57E9AFEE}
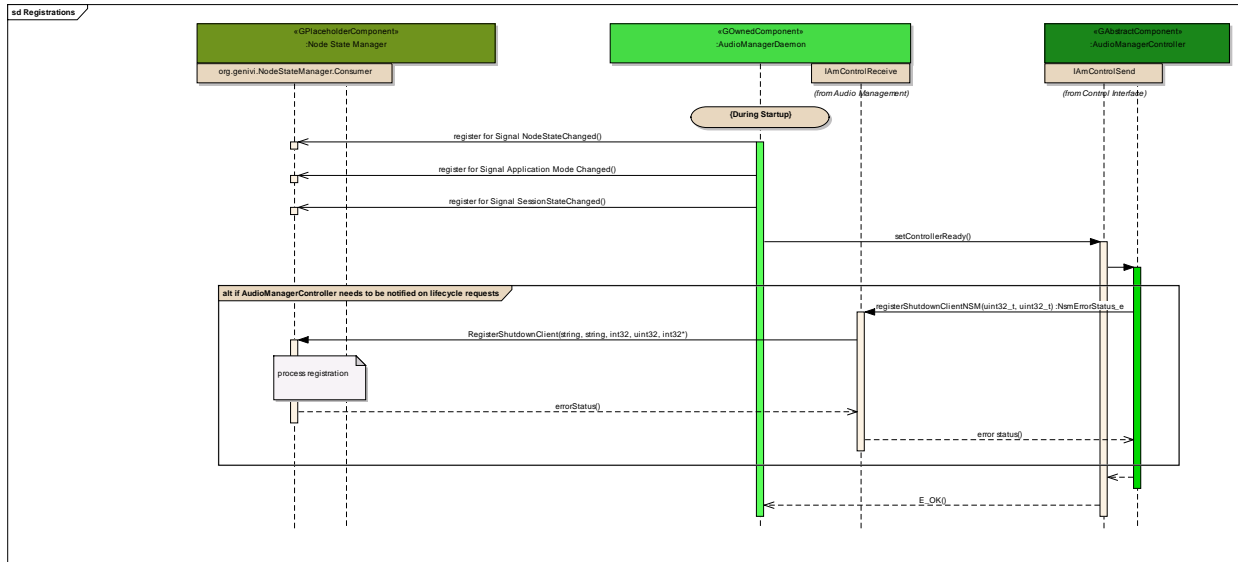
Figure: 122

SessionStateChange - *(Interaction diagram)*

Created By: q232968 *on* 28.01.2013
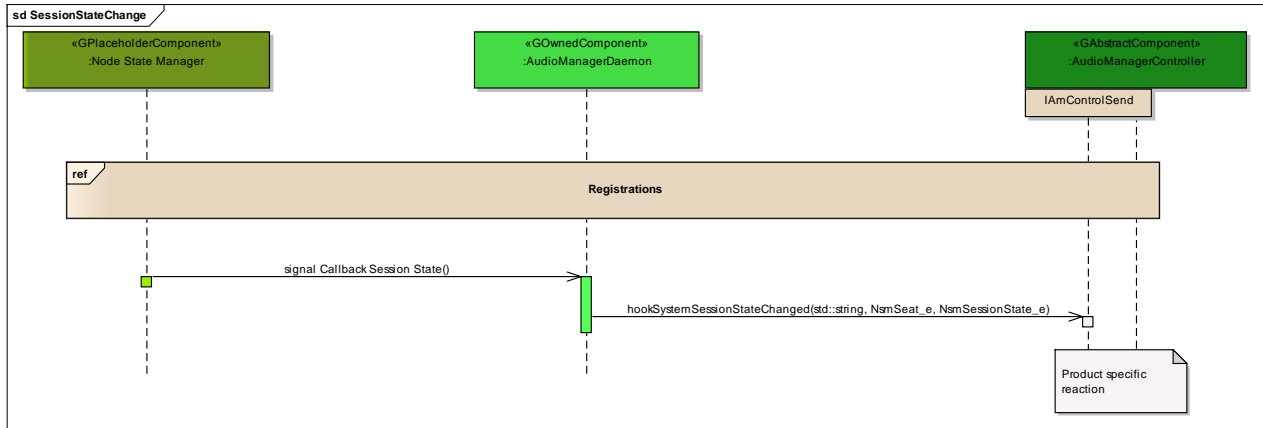
Figure: 123

NodeStateChange - *(Interaction diagram)*
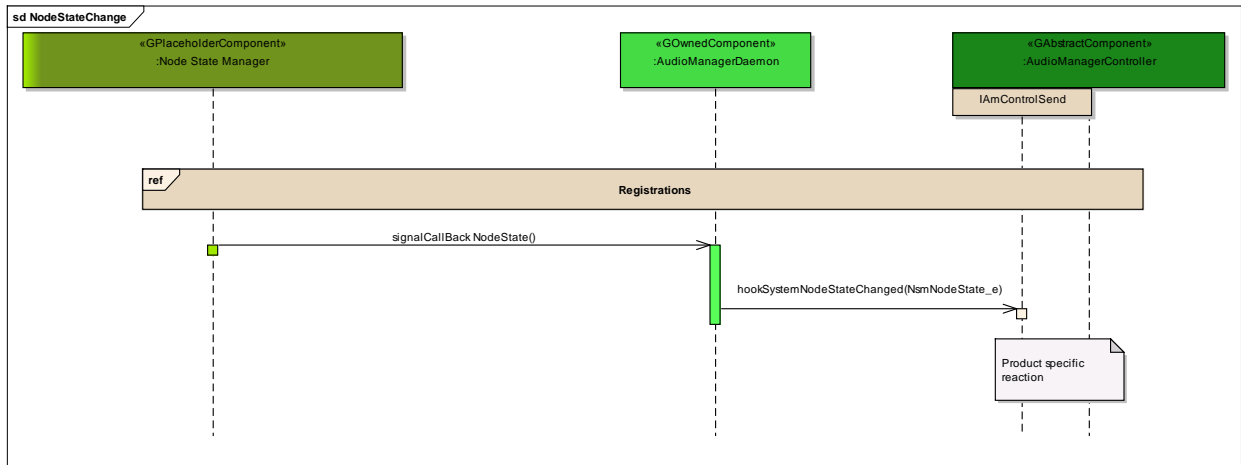
Figure: 124

ApplicationModeChange - *(Interaction diagram)*
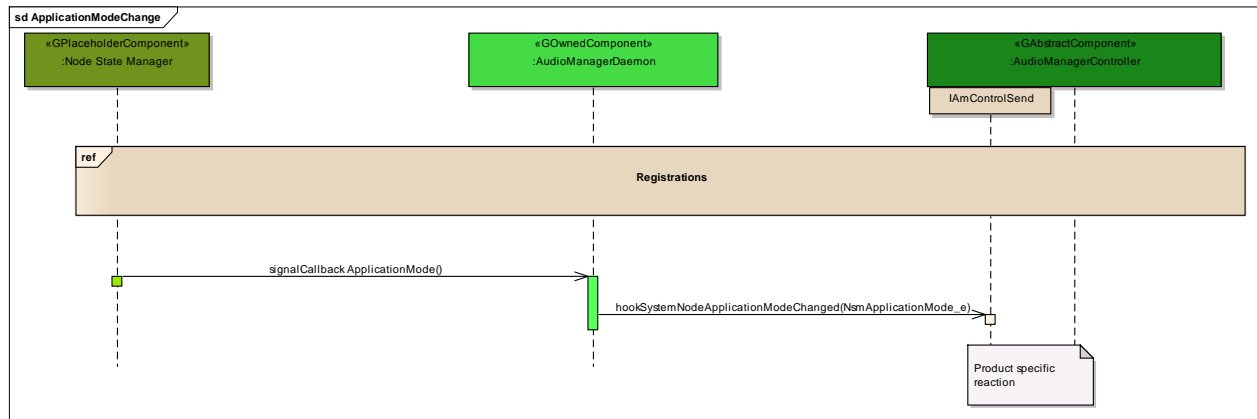
Figure: 125

<anonymous>

Type: **GPlaceholderComponent**
Status: Proposed. Version 1.0. Phase 1.0.
Package: NodeStateManagement   *Keywords:*
Detail: *Created on 28.01.2013. Last modified on 28.01.2013.*
GUID: {5DDE5F17-E368-4b30-8B4E-5B0EEBE27BC0}

Custom Properties

- isIndirectlyInstantiated = True

Connections

Connector	Source	Target	Notes
Dependency Source -> Destination	Public <anonymous>	Public org.genivi.NodeStateManager.LifeCycleConsumer	
Sequence LifecycleRequest Source -> Destination	Public <anonymous>	Public org.genivi.NodeStateManager.LifeCycleConsumer	
Sequence signal	Public	Public	

Connector	Source	Target	Notes
Callback Session State Source -> Destination	\<anonymous>	\<anonymous>	
Sequence signalCallback ApplicationMode Source -> Destination	Public \<anonymous>	Public \<anonymous>	
Sequence signalCallBack NodeState Source -> Destination	Public \<anonymous>	Public \<anonymous>	
Sequence errorStatus Source -> Destination	Public org.genivi.NodeStateManager.LifeCycleConsumer	Public \<anonymous>	
Sequence Source -> Destination	Public org.genivi.NodeStateManager.Consumer	Public \<anonymous>	

Embedded Elements

Element	Detail	Notes
ProvidedInterface com.contiautomotive.NodeStateManager.LifecycleControl	Version: 1.0	
ProvidedInterface org.genivi.NodeStateManager.Consumer	Version: 1.0	

Registrations

Type: **InteractionOccurrence**
Status: Proposed. Version 1.0. Phase 1.0.
Package: NodeStateManagement *Keywords:*
Detail: Created on 28.01.2013. Last modified on 28.01.2013.
GUID: {6E5C065F-23CA-40ba-8BB1-48656F92D3E1}

Connector	Source	Target	Notes
NoteLink	Public <anonymous>	Public Registrations	

process request

Type:	**Sequence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	NodeStateManagement *Keywords:*
Detail:	Created on 28.01.2013. Last modified on 28.01.2013.
GUID:	{A5C7E070-C750-411d-9DE8-4191A6FC1481}

Connections

Connector	Source	Target	Notes
Sequence Source -> Destination	Public IAmControlSend	Public process request	
Sequence LifecycleRequestComplete(uint32, int32, int32*) Source -> Destination	Public process request	Public org.genivi.NodeStateManager.Consumer	

Registrations

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	NodeStateManagement *Keywords:*
Detail:	Created on 28.01.2013. Last modified on 28.01.2013.
GUID:	{7172041F-4C9E-4781-BE1A-1D4D74EE7438}

Registrations

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	NodeStateManagement *Keywords:*
Detail:	*Created on 28.01.2013. Last modified on 28.01.2013.*
GUID:	{CE8A49AF-DB49-487e-807F-D74919201562}

Registrations

Type:	**InteractionOccurrence**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	NodeStateManagement *Keywords:*
Detail:	*Created on 28.01.2013. Last modified on 28.01.2013.*
GUID:	{8CC94B98-ED15-4012-946A-291338803137}

if AudioManagerController needs to be notified on lifecycle requests

Type:	**InteractionFragment**
Status:	Proposed. Version 1.0. Phase 1.0.
Package:	NodeStateManagement *Keywords:*
Detail:	*Created on 28.01.2013. Last modified on 28.01.2013.*
GUID:	{6E13D1AC-7A84-4a91-8E68-12F1FFC9060D}

10 Headers

10.1 ControlReceive

```
/**

 * Copyright (C) 2012, BMW AG

 *

 * This file is part of GENIVI Project AudioManager.

 *

 * Contributions are licensed to the GENIVI Alliance under one or more

 * Contribution License Agreements.

 *

 * \copyright

 * This Source Code Form is subject to the terms of the

 * Mozilla Public License, v. 2.0. If a  copy of the MPL was not distributed with

 * this file, You can obtain one at http://mozilla.org/MPL/2.0/.

 *

 *

 * \author Christian Mueller, christian.linke@bmw.de BMW 2011,2012

 *

 * \file

 * For further information see http://www.genivi.org/.

 *

 * THIS CODE HAS BEEN GENERATED BY ENTERPRISE ARCHITECT GENIVI MODEL. PLEASE CHANGE ONLY IN ENTERPRISE
ARCHITECT AND GENERATE AGAIN

 */

#if !defined(EA_D485A61C_B7C5_4803_B4BD_2AD92893E9CA__INCLUDED_)

#define EA_D485A61C_B7C5_4803_B4BD_2AD92893E9CA__INCLUDED_


#include <vector>

#include <string>

#include "audiomanagertypes.h"

namespace am {

class CAmSocketHandler;
```

```
}


#include "audiomanagertypes.h"
#include "projecttypes.h"
#include "NodeStateManager.h"

#define ControlReceiveVersion "2.0"
namespace am {
        /**
         * This interface gives access to all important functions of the audiomanager that are used by
the AudioManagerController
         * to control the system.
         * There are two rules that have to be kept in mind when implementing against this interface:\n
         * \warning
         * 1. CALLS TO THIS INTERFACE ARE NOT THREAD SAFE !!!! \n
         * 2. YOU MAY NOT CALL THE CALLING INTERFACE DURING AN SYNCHRONOUS OR ASYNCHRONOUS CALL THAT
EXPECTS A RETURN VALUE.\n
         * \details
         * Violation these rules may lead to unexpected behavior! Nevertheless you can implement thread
safe by using the deferred-
         * call pattern described on the wiki which also helps to implement calls that are forbidden.\n
         * For more information, please check CAmSerializer
         * @author Christian Mueller
         * @created 17-Jan-2013 10:00:24
         */
        class IAmControlReceive
        {

        public:
                IAmControlReceive() {

                }

                virtual ~IAmControlReceive() {

                }

                /**
                 * This function returns the version of the interface
                 *
                 * @param version
                 */
                virtual void getInterfaceVersion(std::string& version) const =0;
                /**
                 * calculates a route from source to sink.
                 * @return E_OK on success, E_UNKNOWN on error
                 *
                 * @param onlyfree     if true return only routes which use gateways that are not in use
at the moment
                 * @param sourceID
                 * @param sinkID
                 * @param returnList    this is a list of routes that are possible to take. unsorted! The
longest could be first.
                 * In case not route can be found, the list will return empty.
                 */
                virtual am_Error_e getRoute(const bool onlyfree, const am_sourceID_t sourceID, const
am_sinkID_t sinkID, std::vector<am_Route_s>& returnList) =0;
                /**
                 * With this function, elementary connects can be triggered by the controller.
                 * @return E_OK on success, E_UNKNOWN on error, E_WRONG_FORMAT of connectionFormats do
not match, E_NO_CHANGE if the
                 * desired connection is already build up
                 *
                 * @param handle
                 * @param connectionID
                 * @param format
                 * @param sourceID
                 * @param sinkID
                 */
```

```
                virtual am_Error_e connect(am_Handle_s& handle, am_connectionID_t& connectionID, const
am_ConnectionFormat_e format, const am_sourceID_t sourceID, const am_sinkID_t sinkID) =0;
                /**
                 * is used to disconnect a connection
                 * @return E_OK on success, E_UNKNOWN on error, E_NON_EXISTENT if connection was not
found, E_NO_CHANGE if no change is
                 * neccessary
                 *
                 * @param handle
                 * @param connectionID
                 */
                virtual am_Error_e disconnect(am_Handle_s& handle, const am_connectionID_t connectionID)
=0;
                /**
                 * triggers a cross fade.
                 * @return E_OK on success, E_UNKNOWN on error E_NO_CHANGE if no change is neccessary
                 *
                 * @param handle
                 * @param hotSource    this is the source that is going to be the active one after the
fading
                 * @param crossfaderID
                 * @param rampType
                 * @param rampTime     rampTime 0 means default value
                 */
                virtual am_Error_e crossfade(am_Handle_s& handle, const am_HotSink_e hotSource, const
am_crossfaderID_t crossfaderID, const am_RampType_e rampType, const am_time_t rampTime) =0;
                /**
                 * with this method, all actions that have a handle assigned can be stopped.
                 * @return E_OK on success, E_UNKNOWN on error
                 *
                 * @param handle    the handle of the action to be stopped
                 */
                virtual am_Error_e abortAction(const am_Handle_s handle) =0;
                /**
                 * this method sets a source state for a source. This function will trigger the callback
cbAckSetSourceState
                 * @return E_OK on success, E_NO_CHANGE if the desired value is already correct,
E_UNKNOWN on error, E_NO_CHANGE if no
                 * change is neccessary
                 *
                 * @param handle
                 * @param sourceID
                 * @param state
                 */
                virtual am_Error_e setSourceState(am_Handle_s& handle, const am_sourceID_t sourceID,
const am_SourceState_e state) =0;
                /**
                 * with this function, setting of sinks volumes is done. The behavior of the volume set
is depended on the given ramp and
                 * time information.
                 * This function is not only used to ramp volume, but also to mute and direct set the
level. Exact behavior is depended on
                 * the selected mute ramps.
                 * @return E_OK on success, E_NO_CHANGE if the volume is already on the desired value,
E_OUT_OF_RANGE is the volume is out
                 * of range, E_UNKNOWN on every other error.
                 *
                 * @param handle
                 * @param sinkID
                 * @param volume
                 * @param ramp
                 * @param time
                 */
                virtual am_Error_e setSinkVolume(am_Handle_s& handle, const am_sinkID_t sinkID, const
am_volume_t volume, const am_RampType_e ramp, const am_time_t time) =0;
                /**
                 * with this function, setting of source volumes is done. The behavior of the volume set
is depended on the given ramp and
                 * time information.
                 * This function is not only used to ramp volume, but also to mute and direct set the
level. Exact behavior is depended on
```

```
                * the selected mute ramps.
                * @return E_OK on success, E_NO_CHANGE if the volume is already on the desired value,
E_OUT_OF_RANGE is the volume is out
                * of range, E_UNKNOWN on every other error.
                *
                * @param handle
                * @param sourceID
                * @param volume
                * @param rampType
                * @param time
                */
               virtual am_Error_e setSourceVolume(am_Handle_s& handle, const am_sourceID_t sourceID,
const am_volume_t volume, const am_RampType_e rampType, const am_time_t time) =0;
               /**
                * is used to set several sinkSoundProperties at a time
                * @return E_OK on success, E_UNKNOWN on error, E_OUT_OF_RANGE  if property is out of
range, E_NO_CHANGE if no change is
                * neccessary
                *
                * @param handle
                * @param sinkID
                * @param soundProperty
                */
               virtual am_Error_e setSinkSoundProperties(am_Handle_s& handle, const am_sinkID_t sinkID,
const std::vector<am_SoundProperty_s>& soundProperty) =0;
               /**
                * is used to set sinkSoundProperties
                * @return E_OK on success, E_UNKNOWN on error, E_OUT_OF_RANGE  if property is out of
range, E_NO_CHANGE if no change is
                * neccessary
                *
                * @param handle
                * @param sinkID
                * @param soundProperty
                */
               virtual am_Error_e setSinkSoundProperty(am_Handle_s& handle, const am_sinkID_t sinkID,
const am_SoundProperty_s& soundProperty) =0;
               /**
                * is used to set several SourceSoundProperties at a time
                * @return E_OK on success, E_UNKNOWN on error, E_OUT_OF_RANGE  if property is out of
range. E_NO_CHANGE if no change is
                * neccessary
                *
                * @param handle
                * @param sourceID
                * @param soundProperty
                */
               virtual am_Error_e setSourceSoundProperties(am_Handle_s& handle, const am_sourceID_t
sourceID, const std::vector<am_SoundProperty_s>& soundProperty) =0;
               /**
                * is used to set sourceSoundProperties
                * @return E_OK on success, E_UNKNOWN on error, E_OUT_OF_RANGE  if property is out of
range. E_NO_CHANGE if no change is
                * neccessary
                *
                * @param handle
                * @param sourceID
                * @param soundProperty
                */
               virtual am_Error_e setSourceSoundProperty(am_Handle_s& handle, const am_sourceID_t
sourceID, const am_SoundProperty_s& soundProperty) =0;
               /**
                * sets the domain state of a domain
                * @return E_OK on success, E_UNKNOWN on error, E_NO_CHANGE if no change is neccessary
                *
                * @param domainID
                * @param domainState
                */
               virtual am_Error_e setDomainState(const am_domainID_t domainID, const am_DomainState_e
domainState) =0;
               /**
```

```
                * enters a domain in the database, creates and ID
                * @return E_OK on success, E_ALREADY_EXISTENT if the ID or name is already in the
database, E_DATABASE_ERROR if the
                * database had an error
                *
                * @param domainData    domainID in am_Domain_s must be 0 here
                * @param domainID
                */
               virtual am_Error_e enterDomainDB(const am_Domain_s& domainData, am_domainID_t& domainID)
=0;
               /**
                * enters a mainconnection in the database, creates and ID
                * @return E_OK on success, E_DATABASE_ERROR if the database had an error
                *
                * @param mainConnectionData    the MainConnectionID is omitted since it is created
during the registration
                * @param connectionID
                */
               virtual am_Error_e enterMainConnectionDB(const am_MainConnection_s& mainConnectionData,
am_mainConnectionID_t& connectionID) =0;
               /**
                * enters a sink in the database.
                * The sinkID in am_Sink_s shall be 0 in case of a dynamic added source A sinkID greater
than 100 will be assigned. If a
                * specific sinkID with a value <100 is given, the given value will be used. This is for
a static setup where the ID's are
                * predefined.
                * @return E_OK on success, E_ALREADY_EXISTENT if the ID or name is already in the
database, E_DATABASE_ERROR if the
                * database had an error
                *
                * @param sinkData    the sinkID will be omitted since it is created during the
registration
                * @param sinkID
                */
               virtual am_Error_e enterSinkDB(const am_Sink_s& sinkData, am_sinkID_t& sinkID) =0;
               /**
                * enters a crossfader in the database.
                * The crossfaderID in am_Crossfader_s shall be 0 in case of a dynamic added source A
crossfaderID greater than 100 will
                * be assigned. If a specific crossfaderID with a value <100 is given, the given value
will be used. This is for a static
                * setup where the ID's are predefined.
                * @return E_OK on success, E_ALREADY_EXISTENT if the ID or name is already in the
database, E_DATABASE_ERROR if the
                * database had an error
                *
                * @param crossfaderData    the ID in the data will be ignored since it is created during
the registration
                * @param crossfaderID
                */
               virtual am_Error_e enterCrossfaderDB(const am_Crossfader_s& crossfaderData,
am_crossfaderID_t& crossfaderID) =0;
               /**
                * enters a gateway in the database.
                * The gatewayID in am_Gateway_s shall be 0 in case of a dynamic added source A gatewayID
greater than 100 will be
                * assigned. If a specific gatewayID with a value <100 is given, the given value will be
used. This is for a static setup
                * where the ID's are predefined.
                * @return E_OK on success, E_ALREADY_EXISTENT if the ID or name is already in the
database, E_DATABASE_ERROR if the
                * database had an error
                *
                * @param gatewayData    In a fixed setup, the gatewayID must be below 100. In a dynamic
setup, the gatewayID shall be 0!
                * listSourceFormats and listSinkFormats are empty at registration time. Values are taken
over when sources and sinks are
                * registered.
                * @param gatewayID
                */
```

```
            virtual am_Error_e enterGatewayDB(const am_Gateway_s& gatewayData, am_gatewayID_t&
gatewayID) =0;
            /**
             * enters a source in the database.
             * The sourceID in am_Source_s shall be 0 in case of a dynamic added source A sourceID
greater than 100 will be assigned.
             * If a specific sourceID with a value <100 is given, the given value will be used. This
is for a static setup where the
             * ID's are predefined.
             * @return E_OK on success, E_ALREADY_EXISTENT if the ID or name is already in the
database, E_DATABASE_ERROR if the
             * database had an error
             *
             * @param sourceData    sourceID is omitted here since it is created during registration
             * @param sourceID
             */
            virtual am_Error_e enterSourceDB(const am_Source_s& sourceData, am_sourceID_t& sourceID)
=0;
            /**
             * Enters a sourceClass into the database.
             * @return E_OK on success, E_ALREADY_EXISTENT if the ID or name is already in the
database, E_DATABASE_ERROR if the
             * database had an error
             *
             * @param sinkClass    IDs given with the SinkCLassID are valid below 100 (static). If
given as 0, sinkClass will be
             * enterd dynamic and ID above 100 will be created.
             * @param sinkClassID
             */
            virtual am_Error_e enterSinkClassDB(const am_SinkClass_s& sinkClass, am_sinkClass_t&
sinkClassID) =0;
            /**
             * Enters a sourceClass into the database.
             * The sourceClassID in am_sourceClass_s shall be 0 in case of a dynamic added source A
sourceClassID greater than 100
             * will be assigned. If a specific sourceClassID with a value <100 is given, the given
value will be used. This is for a
             * static setup where the ID's are predefined.
             * @return E_OK on success, E_ALREADY_EXISTENT if the ID or name is already in the
database, E_DATABASE_ERROR if the
             * database had an error
             *
             * @param sourceClassID
             * @param sourceClass    IDs given with the SourceCLassID are valid below 100 (static).
If given as 0, sourceClass will be
             * enterd dynamic and ID above 100 will be created.
             */
            virtual am_Error_e enterSourceClassDB(am_sourceClass_t& sourceClassID, const
am_SourceClass_s& sourceClass) =0;
            /**
             * changes class information of a sinkclass.
             * The properties will overwrite the values of the sinkClassID given in the sinkClass.
             * It is the duty of the controller to check if the property is valid. If it does not
exist, the daemon will not return an
             * error.
             * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if sinkClassID was
not found.
             *
             * @param sinkClass    if a SourceClass ID other than 0 is given, the classInformation
replaces the existing one. In case
             * of ClassID 0, a new classID is created and returned
             */
            virtual am_Error_e changeSinkClassInfoDB(const am_SinkClass_s& sinkClass) =0;
            /**
             * changes class information of a sourceClass.
             * The properties will overwrite the values of the sourceClassID given in the
sourceClass.
             * It is the duty of the controller to check if the property is valid. If it does not
exist, the daemon will not return an
             * error.
             * @return E_OK on success, E_DATABASE_ERROR on error and E_NON_EXISTENT if the ClassID
```

```
does not exist.
                 *
                 * @param sourceClass    if a SourceClass ID other than 0 is given, the classInformation
replaces the existing one. In
                 * case of ClassID 0, a new classID is created and returned
                 */
                virtual am_Error_e changeSourceClassInfoDB(const am_SourceClass_s& sourceClass) =0;
                /**
                 * This function is used to enter the system Properties into the database.
                 * All entries in the database will be erased before entering the new List. It should
only be called once at system
                 * startup.
                 * @return E_OK on success,  E_DATABASE_ERROR if the database had an error
                 *
                 * @param listSystemProperties
                 */
                virtual am_Error_e enterSystemPropertiesListDB(const std::vector<am_SystemProperty_s>&
listSystemProperties) =0;
                /**
                 * changes the mainConnectionState of MainConnection
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if mainconnection
                 *
                 * @param mainconnectionID
                 * @param listConnectionID
                 */
                virtual am_Error_e changeMainConnectionRouteDB(const am_mainConnectionID_t
mainconnectionID, const std::vector<am_connectionID_t>& listConnectionID) =0;
                /**
                 * changes the mainConnectionState of MainConnection
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if mainconnection
                 *
                 * @param mainconnectionID
                 * @param connectionState
                 */
                virtual am_Error_e changeMainConnectionStateDB(const am_mainConnectionID_t
mainconnectionID, const am_ConnectionState_e connectionState) =0;
                /**
                 * changes the sink volume of a sink
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if sink was not
found
                 *
                 * @param mainVolume
                 * @param sinkID
                 */
                virtual am_Error_e changeSinkMainVolumeDB(const am_mainVolume_t mainVolume, const
am_sinkID_t sinkID) =0;
                /**
                 * changes the availablility of a sink
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if sink was not
found
                 *
                 * @param availability
                 * @param sinkID
                 */
                virtual am_Error_e changeSinkAvailabilityDB(const am_Availability_s& availability, const
am_sinkID_t sinkID) =0;
                /**
                 * changes the domainstate of a domain
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if domain was not
found
                 *
                 * @param domainState
                 * @param domainID
                 */
                virtual am_Error_e changDomainStateDB(const am_DomainState_e domainState, const
am_domainID_t domainID) =0;
                /**
                 * changes the mute state of a sink
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if sink was not
found
                 *
```

```
                    * @param muteState
                    * @param sinkID
                    */
                   virtual am_Error_e changeSinkMuteStateDB(const am_MuteState_e muteState, const
am_sinkID_t sinkID) =0;
                   /**
                    * changes the mainsinksoundproperty of a sink
                    * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if sink was not
found
                    *
                    * @param soundProperty
                    * @param sinkID
                    */
                   virtual am_Error_e changeMainSinkSoundPropertyDB(const am_MainSoundProperty_s&
soundProperty, const am_sinkID_t sinkID) =0;
                   /**
                    * changes the mainsourcesoundproperty of a sink
                    * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if source was not
found
                    *
                    * @param soundProperty
                    * @param sourceID
                    */
                   virtual am_Error_e changeMainSourceSoundPropertyDB(const am_MainSoundProperty_s&
soundProperty, const am_sourceID_t sourceID) =0;
                   /**
                    * changes the availablility of a source
                    * @return E_OK on success, E_DATABASE_ERROR  on error, E_NON_EXISTENT if source was not
found
                    *
                    * @param availability
                    * @param sourceID
                    */
                   virtual am_Error_e changeSourceAvailabilityDB(const am_Availability_s& availability,
const am_sourceID_t sourceID) =0;
                   /**
                    * changes a systemProperty
                    * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if property was not
found
                    *
                    * @param property
                    */
                   virtual am_Error_e changeSystemPropertyDB(const am_SystemProperty_s& property) =0;
                   /**
                    * removes a mainconnection from the DB
                    * @return E_OK on success, E_NON_EXISTENT if main connection was not found,
E_DATABASE_ERROR if the database had an error
                    *
                    * @param mainConnectionID
                    */
                   virtual am_Error_e removeMainConnectionDB(const am_mainConnectionID_t mainConnectionID)
=0;
                   /**
                    * removes a sink from the DB
                    * @return E_OK on success, E_NON_EXISTENT if sink was not found, E_DATABASE_ERROR if the
database had an error
                    *
                    * @param sinkID
                    */
                   virtual am_Error_e removeSinkDB(const am_sinkID_t sinkID) =0;
                   /**
                    * removes a source from the DB
                    * @return E_OK on success, E_NON_EXISTENT if source was not found, E_DATABASE_ERROR if
the database had an error
                    *
                    * @param sourceID
                    */
                   virtual am_Error_e removeSourceDB(const am_sourceID_t sourceID) =0;
                   /**
                    * removes a gateway from the DB
                    * @return E_OK on success, E_NON_EXISTENT if gateway was not found, E_DATABASE_ERROR if
```

```
the database had an error
                    *
                    * @param gatewayID
                    */
              virtual am_Error_e removeGatewayDB(const am_gatewayID_t gatewayID) =0;
              /**
                    * removes a crossfader from the DB
                    * @return E_OK on success, E_NON_EXISTENT if crossfader was not found, E_DATABASE_ERROR
if the database had an error
                    *
                    * @param crossfaderID
                    */
              virtual am_Error_e removeCrossfaderDB(const am_crossfaderID_t crossfaderID) =0;
              /**
                    * removes a domain from the DB
                    * @return E_OK on success, E_NON_EXISTENT if domain was not found, E_DATABASE_ERROR if
the database had an error
                    *
                    * @param domainID
                    */
              virtual am_Error_e removeDomainDB(const am_domainID_t domainID) =0;
              /**
                    * removes a domain from the DB
                    * @return E_OK on success, E_NON_EXISTENT if domain was not found, E_DATABASE_ERROR if
the database had an error
                    *
                    * @param sinkClassID
                    */
              virtual am_Error_e removeSinkClassDB(const am_sinkClass_t sinkClassID) =0;
              /**
                    * removes a domain from the DB
                    * @return E_OK on success, E_NON_EXISTENT if domain was not found, E_DATABASE_ERROR if
the database had an error
                    *
                    * @param sourceClassID
                    */
              virtual am_Error_e removeSourceClassDB(const am_sourceClass_t sourceClassID) =0;
              /**
                    * returns the ClassInformation of a source
                    * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if source was not
found
                    *
                    * @param sourceID
                    * @param classInfo
                    */
              virtual am_Error_e getSourceClassInfoDB(const am_sourceID_t sourceID, am_SourceClass_s&
classInfo) const =0;
              /**
                    * returns the ClassInformation of a sink
                    * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if sink was not
found
                    *
                    * @param sinkID
                    * @param sinkClass
                    */
              virtual am_Error_e getSinkClassInfoDB(const am_sinkID_t sinkID, am_SinkClass_s&
sinkClass) const =0;
              /**
                    * returns the sinkData of a sink
                    * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if sink was not
found
                    *
                    * @param sinkID
                    * @param sinkData
                    */
              virtual am_Error_e getSinkInfoDB(const am_sinkID_t sinkID, am_Sink_s& sinkData) const =0;
              /**
                    * returns the sourcekData of a source
                    * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if sink was not
found
                    *
```

```
                 * @param sourceID
                 * @param sourceData
                 */
                virtual am_Error_e getSourceInfoDB(const am_sourceID_t sourceID, am_Source_s& sourceData)
const =0;
                /**
                 * return source and sink of a gateway
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if gateway was not
found
                 *
                 * @param gatewayID
                 * @param gatewayData
                 */
                virtual am_Error_e getGatewayInfoDB(const am_gatewayID_t gatewayID, am_Gateway_s&
gatewayData) const =0;
                /**
                 * returns sources and the sink of a crossfader
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if crossfader was
not found
                 *
                 * @param crossfaderID
                 * @param crossfaderData
                 */
                virtual am_Error_e getCrossfaderInfoDB(const am_crossfaderID_t crossfaderID,
am_Crossfader_s& crossfaderData) const =0;
                /**
                 * returns sources and the sink of a crossfader
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if crossfader was
not found
                 *
                 * @param mainConnectionID
                 * @param mainConnectionData
                 */
                virtual am_Error_e getMainConnectionInfoDB(const am_mainConnectionID_t mainConnectionID,
am_MainConnection_s& mainConnectionData) const =0;
                /**
                 * returns all sinks of a domain
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if domain was not
found
                 *
                 * @param domainID
                 * @param listSinkID
                 */
                virtual am_Error_e getListSinksOfDomain(const am_domainID_t domainID,
std::vector<am_sinkID_t>& listSinkID) const =0;
                /**
                 * returns all source of a domain
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if domain was not
found
                 *
                 * @param domainID
                 * @param listSourceID
                 */
                virtual am_Error_e getListSourcesOfDomain(const am_domainID_t domainID,
std::vector<am_sourceID_t>& listSourceID) const =0;
                /**
                 * returns all crossfaders of a domain
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if domain was not
found
                 *
                 * @param domainID
                 * @param listCrossfadersID
                 */
                virtual am_Error_e getListCrossfadersOfDomain(const am_domainID_t domainID,
std::vector<am_crossfaderID_t>& listCrossfadersID) const =0;
                /**
                 * returns all gateways of a domain
                 * @return E_OK on success, E_DATABASE_ERROR on error, E_NON_EXISTENT if domain was not
found
                 *
                 * @param domainID
```

```
                    * @param listGatewaysID
                    */
                   virtual am_Error_e getListGatewaysOfDomain(const am_domainID_t domainID,
std::vector<am_gatewayID_t>& listGatewaysID) const =0;
                   /**
                    * returns a complete list of all MainConnections
                    * @return E_OK on success, E_DATABASE_ERROR on error
                    *
                    * @param listMainConnections
                    */
                   virtual am_Error_e getListMainConnections(std::vector<am_MainConnection_s>&
listMainConnections) const =0;
                   /**
                    * returns a complete list of all domains
                    * @return E_OK on success, E_DATABASE_ERROR on error
                    *
                    * @param listDomains
                    */
                   virtual am_Error_e getListDomains(std::vector<am_Domain_s>& listDomains) const =0;
                   /**
                    * returns a complete list of all Connections
                    * @return E_OK on success, E_DATABASE_ERROR on error
                    *
                    * @param listConnections
                    */
                   virtual am_Error_e getListConnections(std::vector<am_Connection_s>& listConnections)
const =0;
                   /**
                    * returns a list of all sinks
                    * @return E_OK on success, E_DATABASE_ERROR on error
                    *
                    * @param listSinks
                    */
                   virtual am_Error_e getListSinks(std::vector<am_Sink_s>& listSinks) const =0;
                   /**
                    * returns a list of all sources
                    * @return E_OK on success, E_DATABASE_ERROR on error
                    *
                    * @param listSources
                    */
                   virtual am_Error_e getListSources(std::vector<am_Source_s>& listSources) const =0;
                   /**
                    * returns a list of all source classes
                    * @return E_OK on success, E_DATABASE_ERROR on error
                    *
                    * @param listSourceClasses
                    */
                   virtual am_Error_e getListSourceClasses(std::vector<am_SourceClass_s>& listSourceClasses)
const =0;
                   /**
                    * returns a list of all handles
                    * @return E_OK on success, E_DATABASE_ERROR on error
                    *
                    * @param listHandles
                    */
                   virtual am_Error_e getListHandles(std::vector<am_Handle_s>& listHandles) const =0;
                   /**
                    * returns a list of all crossfaders
                    * @return E_OK on success, E_DATABASE_ERROR on error
                    *
                    * @param listCrossfaders
                    */
                   virtual am_Error_e getListCrossfaders(std::vector<am_Crossfader_s>& listCrossfaders)
const =0;
                   /**
                    * returns a list of  gateways
                    * @return E_OK on success, E_DATABASE_ERROR on error
                    *
                    * @param listGateways
                    */
                   virtual am_Error_e getListGateways(std::vector<am_Gateway_s>& listGateways) const =0;
```

```
            /**
             * returns a list of all sink classes
             * @return E_OK on success, E_DATABASE_ERROR on error
             *
             * @param listSinkClasses
             */
            virtual am_Error_e getListSinkClasses(std::vector<am_SinkClass_s>& listSinkClasses) const
=0;
            /**
             * returns the list of SystemProperties
             *
             * @param listSystemProperties
             */
            virtual am_Error_e getListSystemProperties(std::vector<am_SystemProperty_s>&
listSystemProperties) const =0;
            /**
             * sets the command interface to ready. Will send setCommandReady to each of the plugins.
The corresponding answer is
             * confirmCommandReady.
             */
            virtual void setCommandReady() =0;
            /**
             * sets the command interface into the rundown state. Will send setCommandRundown to each
of the plugins. The
             * corresponding answer is confirmCommandRundown.
             */
            virtual void setCommandRundown() =0;
            /**
             * sets the routinginterface to  ready. Will send the command  setRoutingReady to each of
the plugins. The related answer
             * is confirmRoutingReady.
             */
            virtual void setRoutingReady() =0;
            /**
             * sets the routinginterface to the rundown state. Will send the command
setRoutingRundown to each of the plugins. The
             * related answer is confirmRoutingRundown.
             */
            virtual void setRoutingRundown() =0;
            /**
             * acknowledges the setControllerReady call.
             *
             * @param error    E_OK if the ready command succeeded. E_UNKNOWN if an error happened.
             */
            virtual void confirmControllerReady(const am_Error_e error) =0;
            /**
             * Acknowledges the setControllerRundown call.
             *
             * @param error    E_OK if the ready command succeeded. E_UNKNOWN if an error happened.
             * If an error !=E_OK is returned than the AudioManager will terminate with an exception.
             * If E_OK is returned, the AudioManager will exit clean
             */
            virtual void confirmControllerRundown(const am_Error_e error) =0;
            /**
             * This function returns the pointer to the socketHandler. This can be used to integrate
socket-based activites like
             * communication with the mainloop of the AudioManager.
             * returns E_OK if pointer is valid, E_UNKNOWN in case AudioManager was compiled without
socketHandler support,
             *
             * @param socketHandler
             */
            virtual am_Error_e getSocketHandler(CAmSocketHandler*& socketHandler) =0;
            /**
             * Change the data of the source.
             *
             * @param sourceID    The sourceID of the source that needs to be changed.
             * @param sourceClassID    If this sourceClassID !=0, the sourceClassID of the source
will be changed.
             * @param listSoundProperties    If the size of the list !=0, the list replaces the
current one.
```

```
                 * @param listConnectionFormats    If the size of the list !=0, the list replaces the
current one.
                 * @param listMainSoundProperties    If the size of the list !=0, the list replaces the
current one.
                 */
                virtual am_Error_e changeSourceDB(const am_sourceID_t sourceID, const am_sourceClass_t
sourceClassID, const std::vector<am_SoundProperty_s>& listSoundProperties, const
std::vector<am_ConnectionFormat_e>& listConnectionFormats, const std::vector<am_MainSoundProperty_s>&
listMainSoundProperties) =0;
                /**
                 * Change the data of the sink.
                 *
                 * @param sinkID    the sinkID of the sink to be changed
                 * @param sinkClassID    If !=0, the sinkClass will replace the current one.
                 * @param listSoundProperties    The list of the sound properties. If The lenght of the
list !=0, the current
                 * soundProperties will be changed.
                 * @param listConnectionFormats    The list of the connectionformats to be changed. They
will only be changed, if length
                 * of the list !=0
                 * @param listMainSoundProperties    The list of the mainsoundproperties that need to be
changed. Will be changed. if the
                 * length of the list !=0
                 */
                virtual am_Error_e changeSinkDB(const am_sinkID_t sinkID, const am_sinkClass_t
sinkClassID, const std::vector<am_SoundProperty_s>& listSoundProperties, const
std::vector<am_ConnectionFormat_e>& listConnectionFormats, const std::vector<am_MainSoundProperty_s>&
listMainSoundProperties) =0;
                /**
                 * changes Gateway Data
                 *
                 * @param gatewayID    the gatewayData to be changed
                 * @param listSourceConnectionFormats    The list of the sourceConnectionFormats. Will be
changed it the lenght of the
                 * list !=0.
                 * @param listSinkConnectionFormats    The list of the sinkConnectionFormats. Will be
changed it the lenght of the list
                 * !=0.
                 * @param convertionMatrix    the convertionmatrix
                 */
                virtual am_Error_e changeGatewayDB(const am_gatewayID_t gatewayID, const
std::vector<am_ConnectionFormat_e>& listSourceConnectionFormats, const
std::vector<am_ConnectionFormat_e>& listSinkConnectionFormats, const std::vector<bool>&
convertionMatrix) =0;
                /**
                 * with this function, setting of multiple volumes at a time is done. The behavior of the
volume set is depended on the
                 * given ramp and time information.
                 * This function is not only used to ramp volume, but also to mute and direct set the
level. Exact behavior is depended on
                 * the selected mute ramps.
                 * @return E_OK on success, E_NO_CHANGE if the volume is already on the desired value,
E_OUT_OF_RANGE is the volume is out
                 * of range, E_UNKNOWN on every other error.
                 *
                 * @param handle
                 * @param listVolumes
                 */
                virtual am_Error_e setVolumes(am_Handle_s& handle, const std::vector<am_Volumes_s>&
listVolumes) =0;
                /**
                 * set a sink notification configuration
                 *
                 * @param handle    the handle that will be assigned for this operation
                 * @param sinkID    the sinkID
                 * @param notificationConfiguration
                 */
                virtual am_Error_e setSinkNotificationConfiguration(am_Handle_s& handle, const
am_sinkID_t sinkID, const am_NotificationConfiguration_s& notificationConfiguration) =0;
                /**
                 * set a source notification configuration
```

```
             *
             * @param handle    the handle that will be assigned for this operation
             * @param sourceID    the sinkID
             * @param notificationConfiguration
             */
            virtual am_Error_e setSourceNotificationConfiguration(am_Handle_s& handle, const
am_sourceID_t sourceID, const am_NotificationConfiguration_s& notificationConfiguration) =0;
            /**
             * Sends out the main notificiation of a sink
             * @return E_OK on success, E_UNKNOWN on error
             *
             * @param sinkID
             * @param notificationPayload    the payload
             */
            virtual void sendMainSinkNotificationPayload(const am_sinkID_t sinkID, const
am_NotificationPayload_s& notificationPayload) =0;
            /**
             * Sends out the main notificiation of a source
             * @return E_OK on success, E_UNKNOWN on error
             *
             * @param sourceID
             * @param notificationPayload    the payload
             */
            virtual void sendMainSourceNotificationPayload(const am_sourceID_t sourceID, const
am_NotificationPayload_s& notificationPayload) =0;
            /**
             * change the mainNotificationConfiguration of a sink
             * @return E_OK when successful, E_DATABASE on error
             *
             * @param sinkID    the sinkID
             * @param mainNotificationConfiguration    the mainNotificationConfiguration
             */
            virtual am_Error_e changeMainSinkNotificationConfigurationDB(const am_sinkID_t sinkID,
const am_NotificationConfiguration_s& mainNotificationConfiguration) =0;
            /**
             * change the mainNotificationConfiguration of a source
             * @return E_OK when successful, E_DATABASE on error
             *
             * @param sourceID    the sourceID
             * @param mainNotificationConfiguration    the mainNotificationConfiguration
             */
            virtual am_Error_e changeMainSourceNotificationConfigurationDB(const am_sourceID_t
sourceID, const am_NotificationConfiguration_s& mainNotificationConfiguration) =0;
            /**
             * Retrieves the Restart Reason Property
             * @return E_OK on success E_UNKNOWN if property could not be retrieved.
             *
             * @param restartReason    the restart reason
             */
            virtual am_Error_e getRestartReasonPropertyNSM(NsmRestartReason_e& restartReason) =0;
            /**
             * Retrieves the shutdown reason property
             * @return E_OK on success E_UNKNOWN if property could not be retrieved.
             *
             * @param ShutdownReason    The shutdown reason
             */
            virtual am_Error_e getShutdownReasonPropertyNSM(NsmShutdownReason_e& ShutdownReason) =0;
            /**
             * Retrieves the running reason
             * @return E_OK on success E_UNKNOWN if property could not be retrieved.
             *
             * @param nsmRunningReason    the running reson
             */
            virtual am_Error_e getRunningReasonPropertyNSM(NsmRunningReason_e& nsmRunningReason) =0;
            /**
             * @return NsmErrorStatus_Ok in case of success
             *
             * @param nsmNodeState    The current node state
             */
            virtual NsmErrorStatus_e getNodeStateNSM(NsmNodeState_e& nsmNodeState) =0;
            /**
```

```
              * Retrieves the current session state
              * @return NsmErrorStatus_Ok on success E_UNKNOWN if property could not be retrieved.
              *
              * @param sessionName    The current session state
              * @param seatID
              * @param sessionState    The sessionState
              */
            virtual NsmErrorStatus_e getSessionStateNSM(const std::string& sessionName, const
NsmSeat_e seatID, NsmSessionState_e& sessionState) =0;
            /**
              * Retrieves the current applicationmode
              * @return NsmErrorStatus_Ok on success E_UNKNOWN if property could not be retrieved.
              *
              * @param applicationMode    The current application Mode
              */
            virtual NsmErrorStatus_e getApplicationModeNSM(NsmApplicationMode_e& applicationMode) =0;
            /**
              * Registers the AudioManager as shutdown client
              * @return NsmErrorStatus_Ok on success
              * The interface for the client will be provided by the AudioManagerDeamon, therefore is
no need to mention it here.
              *
              * @param shutdownMode    The shutdown mode that the controller wants to register for.
              * @param timeoutMs    Max. Timeout to wait for response from shutdown client in ms
              */
            virtual NsmErrorStatus_e registerShutdownClientNSM(const uint32_t shutdownMode, const
uint32_t timeoutMs) =0;
            /**
              * unregisters the AudioManager as shutdown client
              * @return NsmErrorStatus_Ok on success
              *
              * @param shutdownMode    Shutdown mode for which client wants to unregister
(NSM_SHUTDOWNTYPE_NORMAL,
              * NSM_SHUTDOWNTYPE_FAST)
              */
            virtual NsmErrorStatus_e unRegisterShutdownClientNSM(const uint32_t shutdownMode) =0;
            /**
              * @return E_OK on success, E_UNKOWN if interface could not be reached
              *
              * @param version
              */
            virtual am_Error_e getInterfaceVersionNSM(uint32_t& version) =0;
            /**
              * Sends out the lifecyclerequest complete to the NSM
              * @return NsmErrorStatus_Ok in case of success
              *
              * @param RequestId    The request ID of the request.
              * @param status    The result of the call to "LifecycleRequest". NsmErrorStatus_Ok:
Request successfully processed.
              * NsmErrorStatus_Error: An error occured while processing the "LifecycleRequest".
              */
            virtual NsmErrorStatus_e sendLifecycleRequestCompleteNSM(const uint32_t RequestId, const
NsmErrorStatus_e status) =0;

      };
}
#endif // !defined(EA_D485A61C_B7C5_4803_B4BD_2AD92893E9CA__INCLUDED_)
```

10.2 ControlSend

```
/**

 * Copyright (C) 2012, BMW AG

 *
```

```
 * This file is part of GENIVI Project AudioManager.
 *
 * Contributions are licensed to the GENIVI Alliance under one or more
 * Contribution License Agreements.
 *
 * \copyright
 * This Source Code Form is subject to the terms of the
 * Mozilla Public License, v. 2.0. If a  copy of the MPL was not distributed with
 * this file, You can obtain one at http://mozilla.org/MPL/2.0/.
 *
 *
 * \author Christian Mueller, christian.linke@bmw.de BMW 2011,2012
 *
 * \file
 * For further information see http://www.genivi.org/.
 *
 * THIS CODE HAS BEEN GENERATED BY ENTERPRISE ARCHITECT GENIVI MODEL. PLEASE CHANGE ONLY IN ENTERPRISE
ARCHITECT AND GENERATE AGAIN
 */
#if !defined(EA_89F9440C_319B_486f_BAB5_91FED9A658C2__INCLUDED_)
#define EA_89F9440C_319B_486f_BAB5_91FED9A658C2__INCLUDED_


#include <vector>
#include <string>
#include "audiomanagertypes.h"


namespace am {
class IAmControlReceive;
}
#include "IAmControlReceive.h"
#include "audiomanagertypes.h"
#include "NodeStateManager.h"

#define ControlSendVersion "2.0"
namespace am {
        /**
         * This interface is presented by the AudioManager controller.
         * All the hooks represent system events that need to be handled. The callback functions are
used to handle for example
         * answers to function calls on the AudioManagerCoreInterface.
         * There are two rules that have to be kept in mind when implementing against this interface:\n
         * \warning
         * 1. CALLS TO THIS INTERFACE ARE NOT THREAD SAFE !!!! \n
```

```
         * 2. YOU MAY NOT CALL THE CALLING INTERFACE DURING AN SYNCHRONOUS OR ASYNCHRONOUS CALL THAT
EXPECTS A RETURN VALUE.\n
         * \details
         * Violation these rules may lead to unexpected behavior! Nevertheless you can implement thread
safe by using the deferred-
         * call pattern described on the wiki which also helps to implement calls that are forbidden.\n
         * For more information, please check CAmSerializer
         * @author Christian Mueller
         * @created 17-Jan-2013 10:00:25
         */
        class IAmControlSend
        {

        public:
                IAmControlSend() {

                }

                virtual ~IAmControlSend() {

                }

                /**
                 * This function returns the version of the interface
                 * returns E_OK, E_UNKOWN if version is unknown.
                 *
                 * @param version
                 */
                virtual void getInterfaceVersion(std::string& version) const =0;
                /**
                 * Starts up the controller.
                 *
                 * @param controlreceiveinterface    This is a pointer to the ControlReceiveInterface so
that the controller knows to whom
                 * to communicate.
                 */
                virtual am_Error_e startupController(IAmControlReceive* controlreceiveinterface) =0;
                /**
                 * this message is used tell the controller that it should get ready. This message must
be acknowledged via
                 * confirmControllerReady.
                 */
                virtual void setControllerReady() =0;
                /**
                 * This message tells the controller that he should prepare everything for the power to
be switched off. This message must
                 * be acknowledged via confirmControllerRundown.
                 * The method will give the signal as integer that was responsible for calling the
setControllerRundown.
                 * This function is called from the signal handler, either direct (when the program is
killed) or from within the mainloop
                 * (if the program is terminated).
                 *
                 * @param signal    The signal NO that was responsible for the rundown. Can be:
                 * SIGINT
                 * SIGQUIT
                 * SIGTERM
                 * SIGHUP
                 */
                virtual void setControllerRundown(const int16_t signal) =0;
                /**
                 * is called when a connection request comes in via the command interface
                 * @return E_OK on success, E_NOT_POSSIBLE on error, E_ALREADY_EXISTENT if already exists
                 *
                 * @param sourceID
                 * @param sinkID
                 * @param mainConnectionID
                 */
                virtual am_Error_e hookUserConnectionRequest(const am_sourceID_t sourceID, const
am_sinkID_t sinkID, am_mainConnectionID_t& mainConnectionID) =0;
                /**
```

```
                * is called when a disconnection request comes in via the command interface
                * @return E_OK on success, E_NOT_POSSIBLE on error, E_NON_EXISTENT if connection does
not exists
                *
                * @param connectionID
                */
            virtual am_Error_e hookUserDisconnectionRequest(const am_mainConnectionID_t connectionID)
=0;
                /**
                * sets a user MainSinkSoundProperty
                * @return E_OK on success, E_OUT_OF_RANGE if out of range, E_UNKNOWN on error
                *
                * @param sinkID
                * @param soundProperty
                */
            virtual am_Error_e hookUserSetMainSinkSoundProperty(const am_sinkID_t sinkID, const
am_MainSoundProperty_s& soundProperty) =0;
                /**
                * sets a user MainSourceSoundProperty
                * @return E_OK on success, E_OUT_OF_RANGE if out of range, E_UNKNOWN on error
                *
                * @param sourceID
                * @param soundProperty
                */
            virtual am_Error_e hookUserSetMainSourceSoundProperty(const am_sourceID_t sourceID, const
am_MainSoundProperty_s& soundProperty) =0;
                /**
                * sets a user SystemProperty
                * @return E_OK on success, E_OUT_OF_RANGE if out of range, E_UNKNOWN on error
                *
                * @param property
                */
            virtual am_Error_e hookUserSetSystemProperty(const am_SystemProperty_s& property) =0;
                /**
                * sets a user volume
                * @return E_OK on success, E_OUT_OF_RANGE if out of range, E_UNKNOWN on error
                *
                * @param SinkID
                * @param newVolume
                */
            virtual am_Error_e hookUserVolumeChange(const am_sinkID_t SinkID, const am_mainVolume_t
newVolume) =0;
                /**
                * sets a user volume as increment
                * @return E_OK on success, E_OUT_OF_RANGE if out of range, E_UNKNOWN on error
                *
                * @param SinkID
                * @param increment    the steps
                */
            virtual am_Error_e hookUserVolumeStep(const am_sinkID_t SinkID, const int16_t increment)
=0;
                /**
                * sets the mute state of a sink
                * @return E_OK on success, E_UNKNOWN on error
                *
                * @param sinkID
                * @param muteState    true=muted
                */
            virtual am_Error_e hookUserSetSinkMuteState(const am_sinkID_t sinkID, const
am_MuteState_e muteState) =0;
                /**
                * is called when a routing adaptor registers its domain
                * @return E_OK on success, E_UNKNOWN on error, E_ALREADY_EXISTENT if already exists
                *
                * @param domainData    ID is omitted here since it has not been created yet
                * @param domainID
                */
            virtual am_Error_e hookSystemRegisterDomain(const am_Domain_s& domainData, am_domainID_t&
domainID) =0;
                /**
                * is called when a routing adaptor wants to derigister a domain
```

```
                 * @return E_OK on success, E_UNKNOWN on error, E_NON_EXISTENT if not found
                 *
                 * @param domainID
                 */
                virtual am_Error_e hookSystemDeregisterDomain(const am_domainID_t domainID) =0;
                /**
                 * is called when a domain registered all the elements
                 *
                 * @param domainID
                 */
                virtual void hookSystemDomainRegistrationComplete(const am_domainID_t domainID) =0;
                /**
                 * is called when a routing adaptor registers a sink
                 * @return E_OK on success, E_UNKNOWN on error, E_ALREADY_EXISTENT if already exists
                 *
                 * @param sinkData    Id is omitted here, since it has not been created yet
                 * @param sinkID
                 */
                virtual am_Error_e hookSystemRegisterSink(const am_Sink_s& sinkData, am_sinkID_t& sinkID)
=0;
                /**
                 * is called when a routing adaptor deregisters a sink
                 * @return E_OK on success, E_UNKNOWN on error, E_NON_EXISTENT if not found
                 *
                 * @param sinkID
                 */
                virtual am_Error_e hookSystemDeregisterSink(const am_sinkID_t sinkID) =0;
                /**
                 * is called when a routing adaptor registers a source
                 * @return E_OK on success, E_UNKNOWN on error, E_ALREADY_EXISTENT if already exists
                 *
                 * @param sourceData    ID is omitted here since it is not yet created
                 * @param sourceID
                 */
                virtual am_Error_e hookSystemRegisterSource(const am_Source_s& sourceData, am_sourceID_t&
sourceID) =0;
                /**
                 * is called when a routing adaptor deregisters a source
                 * @return E_OK on success, E_UNKNOWN on error, E_NON_EXISTENT if not found
                 *
                 * @param sourceID
                 */
                virtual am_Error_e hookSystemDeregisterSource(const am_sourceID_t sourceID) =0;
                /**
                 * is called when a routing adaptor registers a gateway
                 * @return E_OK on success, E_UNKNOWN on error, E_ALREADY_EXISTENT if already exists
                 *
                 * @param gatewayData    gatewayID is not set here since it is not created at this point
of time
                 * @param gatewayID
                 */
                virtual am_Error_e hookSystemRegisterGateway(const am_Gateway_s& gatewayData,
am_gatewayID_t& gatewayID) =0;
                /**
                 * is called when a routing adaptor deregisters a gateway
                 * @return E_OK on success, E_UNKNOWN on error, E_NON_EXISTENT if not found
                 *
                 * @param gatewayID
                 */
                virtual am_Error_e hookSystemDeregisterGateway(const am_gatewayID_t gatewayID) =0;
                /**
                 * is called when a routing adaptor registers a crossfader
                 * @return E_OK on success, E_UNKNOWN on error, E_ALREADY_EXISTENT if already exists
                 *
                 * @param crossfaderData    gatewayID is not set here since it is not created at this
point of time
                 * @param crossfaderID
                 */
                virtual am_Error_e hookSystemRegisterCrossfader(const am_Crossfader_s& crossfaderData,
am_crossfaderID_t& crossfaderID) =0;
                /**
```

```
                       * is called when a routing adaptor deregisters a crossfader
                       * @return E_OK on success, E_UNKNOWN on error, E_NON_EXISTENT if not found
                       *
                       * @param crossfaderID
                       */
                      virtual am_Error_e hookSystemDeregisterCrossfader(const am_crossfaderID_t crossfaderID)
=0;
                      /**
                       * volumeticks. therse are used to indicate volumechanges during a ramp
                       *
                       * @param handle
                       * @param sinkID
                       * @param volume
                       */
                      virtual void hookSystemSinkVolumeTick(const am_Handle_s handle, const am_sinkID_t sinkID,
const am_volume_t volume) =0;
                      /**
                       * volumeticks. therse are used to indicate volumechanges during a ramp
                       *
                       * @param handle
                       * @param sourceID
                       * @param volume
                       */
                      virtual void hookSystemSourceVolumeTick(const am_Handle_s handle, const am_sourceID_t
sourceID, const am_volume_t volume) =0;
                      /**
                       * is called when an low level interrupt changed its state
                       *
                       * @param sourceID
                       * @param interruptState
                       */
                      virtual void hookSystemInterruptStateChange(const am_sourceID_t sourceID, const
am_InterruptState_e interruptState) =0;
                      /**
                       * id called when a sink changed its availability
                       *
                       * @param sinkID
                       * @param availability
                       */
                      virtual void hookSystemSinkAvailablityStateChange(const am_sinkID_t sinkID, const
am_Availability_s& availability) =0;
                      /**
                       * id called when a source changed its availability
                       *
                       * @param sourceID
                       * @param availability
                       */
                      virtual void hookSystemSourceAvailablityStateChange(const am_sourceID_t sourceID, const
am_Availability_s& availability) =0;
                      /**
                       * id called when domainstate was changed
                       *
                       * @param domainID
                       * @param state
                       */
                      virtual void hookSystemDomainStateChange(const am_domainID_t domainID, const
am_DomainState_e state) =0;
                      /**
                       * when early data was received
                       *
                       * @param data
                       */
                      virtual void hookSystemReceiveEarlyData(const std::vector<am_EarlyData_s>& data) =0;
                      /**
                       * this hook provides information about speed changes.
                       * The quantization and sampling rate of the speed can be adjusted at compile time of the
AudioManagerDaemon.
                       *
                       * @param speed
                       */
                      virtual void hookSystemSpeedChange(const am_speed_t speed) =0;
```

```
          /**
           * this hook is fired whenever the timing information of a mainconnection has changed.
           *
           * @param mainConnectionID
           * @param time
           */
          virtual void hookSystemTimingInformationChanged(const am_mainConnectionID_t
mainConnectionID, const am_timeSync_t time) =0;
          /**
           * ack for connect
           *
           * @param handle
           * @param errorID
           */
          virtual void cbAckConnect(const am_Handle_s handle, const am_Error_e errorID) =0;
          /**
           * ack for disconnect
           *
           * @param handle
           * @param errorID
           */
          virtual void cbAckDisconnect(const am_Handle_s handle, const am_Error_e errorID) =0;
          /**
           * ack for crossfading
           *
           * @param handle
           * @param hostsink
           * @param error
           */
          virtual void cbAckCrossFade(const am_Handle_s handle, const am_HotSink_e hostsink, const
am_Error_e error) =0;
          /**
           * ack for sink volume changes
           *
           * @param handle
           * @param volume
           * @param error
           */
          virtual void cbAckSetSinkVolumeChange(const am_Handle_s handle, const am_volume_t volume,
const am_Error_e error) =0;
          /**
           * ack for source volume changes
           *
           * @param handle    the handle that is connected to the volume change
           * @param voulme    the volume after the action ended (the desired volume if everything
went right, the actual one in case
           * of abortion)
           * @param error
           */
          virtual void cbAckSetSourceVolumeChange(const am_Handle_s handle, const am_volume_t
voulme, const am_Error_e error) =0;
          /**
           * ack for setting of source states
           *
           * @param handle
           * @param error
           */
          virtual void cbAckSetSourceState(const am_Handle_s handle, const am_Error_e error) =0;
          /**
           * ack for setting of sourcesoundproperties
           *
           * @param handle
           * @param error
           */
          virtual void cbAckSetSourceSoundProperties(const am_Handle_s handle, const am_Error_e
error) =0;
          /**
           * ack for setting of sourcesoundproperties
           *
           * @param handle
           * @param error
```

```
                */
               virtual void cbAckSetSourceSoundProperty(const am_Handle_s handle, const am_Error_e
error) =0;
               /**
                * ack for setting of sinksoundproperties
                *
                * @param handle
                * @param error
                */
               virtual void cbAckSetSinkSoundProperties(const am_Handle_s handle, const am_Error_e
error) =0;
               /**
                * ack for setting of sinksoundproperties
                *
                * @param handle
                * @param error
                */
               virtual void cbAckSetSinkSoundProperty(const am_Handle_s handle, const am_Error_e error)
=0;
               /**
                * This function is used by the routing algorithm to retrieve a priorized list of
connectionFormats from the Controller.
                * @return E_OK in case of successfull priorisation.
                *
                * @param sourceID    sourceID of source that shall be connected
                * @param sinkID    sinkID of sink that shall be connected
                * @param listRoute    This route is the one the priorized connectionFormats is for.
                * @param listPossibleConnectionFormats    list of possible connectionformats
                * @param listPrioConnectionFormats    the list return with prioos from the controller.
Best choice on first position.
                */
               virtual am_Error_e getConnectionFormatChoice(const am_sourceID_t sourceID, const
am_sinkID_t sinkID, const am_Route_s listRoute, const std::vector<am_ConnectionFormat_e>
listPossibleConnectionFormats, std::vector<am_ConnectionFormat_e>& listPrioConnectionFormats) =0;
               /**
                * confirms the setCommandReady call
                *
                * @param error    E_OK if everything went right. If a plugin returns an error, it will
be forwarded here
                */
               virtual void confirmCommandReady(const am_Error_e error) =0;
               /**
                * confirms the setRoutingReady call
                *
                * @param error    E_OK if everything went right. If a plugin returns an error, it will
be forwarded here
                */
               virtual void confirmRoutingReady(const am_Error_e error) =0;
               /**
                * confirms the setCommandRundown call
                *
                * @param error    E_OK if everything went right. If a plugin returns an error, it will
be forwarded here
                */
               virtual void confirmCommandRundown(const am_Error_e error) =0;
               /**
                * confirms the setRoutingRundown command
                *
                * @param error    E_OK if everything went right. If a plugin returns an error, it will
be forwarded here
                */
               virtual void confirmRoutingRundown(const am_Error_e error) =0;
               /**
                * update form the SinkData
                *
                * @param sinkID    the sinkID
                * @param sinkClassID
                * @param listSoundProperties
                * @param listConnectionFormats
                * @param listMainSoundProperties
                */
```

```
            virtual am_Error_e hookSystemUpdateSink(const am_sinkID_t sinkID, const am_sinkClass_t
sinkClassID, const std::vector<am_SoundProperty_s>& listSoundProperties, const
std::vector<am_ConnectionFormat_e>& listConnectionFormats, const std::vector<am_MainSoundProperty_s>&
listMainSoundProperties) =0;
            /**
             * update from the source Data
             *
             * @param sourceID    the sourceID
             * @param sourceClassID
             * @param listSoundProperties
             * @param listConnectionFormats
             * @param listMainSoundProperties
             */
            virtual am_Error_e hookSystemUpdateSource(const am_sourceID_t sourceID, const
am_sourceClass_t sourceClassID, const std::vector<am_SoundProperty_s>& listSoundProperties, const
std::vector<am_ConnectionFormat_e>& listConnectionFormats, const std::vector<am_MainSoundProperty_s>&
listMainSoundProperties) =0;
            /**
             * updates the Gateway Data
             *
             * @param gatewayID    the gatewayID
             * @param listSourceConnectionFormats
             * @param listSinkConnectionFormats
             * @param convertionMatrix
             */
            virtual am_Error_e hookSystemUpdateGateway(const am_gatewayID_t gatewayID, const
std::vector<am_ConnectionFormat_e>& listSourceConnectionFormats, const
std::vector<am_ConnectionFormat_e>& listSinkConnectionFormats, const std::vector<bool>&
convertionMatrix) =0;
            /**
             * ack for mulitple volume changes
             *
             * @param handle
             * @param listVolumes    the list of volumes with the current status
             * @param error
             */
            virtual void cbAckSetVolumes(const am_Handle_s handle, const std::vector<am_Volumes_s>&
listVolumes, const am_Error_e error) =0;
            /**
             * The acknowledge of the sink notification configuration
             *
             * @param handle    the handle
             * @param error    the error
             */
            virtual void cbAckSetSinkNotificationConfiguration(const am_Handle_s handle, const
am_Error_e error) =0;
            /**
             * The acknowledge of the source notification configuration
             *
             * @param handle    the handle
             * @param error    the error
             */
            virtual void cbAckSetSourceNotificationConfiguration(const am_Handle_s handle, const
am_Error_e error) =0;
            /**
             * new sinkNotification data is there!
             *
             * @param sinkID    the sinkID of the changed notification
             * @param payload    the payload
             */
            virtual void hookSinkNotificationDataChanged(const am_sinkID_t sinkID, const
am_NotificationPayload_s& payload) =0;
            /**
             * new sourceNotification data is there!
             *
             * @param sourceID    the sourceID of the changed notification
             * @param payload    the payload
             */
            virtual void hookSourceNotificationDataChanged(const am_sourceID_t sourceID, const
am_NotificationPayload_s& payload) =0;
            /**
```

```
                 * sets a user MainSinkNotificationConfiguration
                 * @return E_OK on success, E_OUT_OF_RANGE if out of range, E_UNKNOWN on error
                 *
                 * @param sinkID
                 * @param notificationConfiguration
                 */
                virtual am_Error_e hookUserSetMainSinkNotificationConfiguration(const am_sinkID_t sinkID,
const am_NotificationConfiguration_s& notificationConfiguration) =0;
                /**
                 * sets a user MainSourceNotificationConfiguration
                 * @return E_OK on success, E_OUT_OF_RANGE if out of range, E_UNKNOWN on error
                 *
                 * @param sourceID
                 * @param notificationConfiguration
                 */
                virtual am_Error_e hookUserSetMainSourceNotificationConfiguration(const am_sourceID_t
sourceID, const am_NotificationConfiguration_s& notificationConfiguration) =0;
                /**
                 * This hook is connected to the NodeState signal of the NodeStateManager
                 *
                 * @param NodeStateId    The new current state of the node
                 */
                virtual void hookSystemNodeStateChanged(const NsmNodeState_e NodeStateId) =0;
                /**
                 * This hook is connected to the NodeApplicationMode signal of the NodeStateManager
                 *
                 * @param ApplicationModeId    The new ApplicationMode
                 */
                virtual void hookSystemNodeApplicationModeChanged(const NsmApplicationMode_e
ApplicationModeId) =0;
                /**
                 * the sessionstate change was send by the NSM
                 *
                 * @param sessionName
                 * @param seatID
                 * @param sessionStateID
                 */
                virtual void hookSystemSessionStateChanged(const std::string& sessionName, const
NsmSeat_e seatID, const NsmSessionState_e sessionStateID) =0;
                /**
                 * A lifecycle request comes in from the NSM and must be processed.
                 * @return NsmErrorStatus_Ok if successful
                 * The controller has to respond to this message with LifecycleRequestComplete.
                 *
                 * @param Request    the shutdown request from the NSM. One of the types
                 *
                 * NSM_SHUTDOWNTYPE_RUNUP, NSM_SHUTDOWNTYPE_NORMAL
                 * NSM_SHUTDOWNTYPE_FAST
                 *
                 * In order to receive this request, the controller needs to register for this request
first via RegisterShutdownClient
                 * @param RequestId    the requestID
                 */
                virtual NsmErrorStatus_e hookSystemLifecycleRequest(const uint32_t Request, const
uint32_t RequestId) =0;

        };
}
#endif // !defined(EA_89F9440C_319B_486f_BAB5_91FED9A658C2__INCLUDED_)
```