RESTful Network API for Audio Call

Candidate Version 1.0 – 12 Feb 2013

Open Mobile Alliance

OMA-TS-REST_NetAPI_AudioCall-V1_0-20130212-C

Contents

Figures

Tables

1. Scope

This specification defines a RESTful Audio Call API using an HTTP protocol binding, based on the similar API defined in [3GPP 29.199-11].

2. References

2.1 Normative References

[3GPP 29.199-11] 3GPP Technical Specification, "Open Service Access (OSA); Parlay X Web Services; Part 11: Audio Call (Release 8)", URL:http://www.3gpp.org/

[Autho4API_10] "Authorization Framework for Network APIs", Open Mobile Alliance™, OMA-ER-Autho4API-V1_0, URL: http://www.openmobilealliance.org/

[IETF_ACR_draft] "The acr URI for anonymous users", S.Jakobsson, K.Smith, July 2011, URL: http://tools.ietf.org/html/draft-uri-acr-extension-03

[REST_NetAPI_ CallNotif] "RESTful Network API for Call Notification", Open Mobile Alliance™, OMA-TS-REST_NetAPI_CallNotification-V1_0, URL:http://www.openmobilealliance.org/

[REST_NetAPI_Common] "Common definitions for RESTful Network APIs", Open Mobile Alliance™, OMA-TS-REST_NetAPI_Common-V1_0, URL:http://www.openmobilealliance.org/

[REST_SUP_AudioCall] "XML schema for the RESTful Network API for AudioCall", Open Mobile Alliance™, OMA-SUP-XSD-rest_netapi_audiocall-V1.0, URL: http://www.openmobilealliance.org/

[RFC2119] "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997, URL:http://www.ietf.org/rfc/rfc2119.txt

[RFC2616] "Hypertext Transfer Protocol -- HTTP/1.1", R. Fielding et. al, January 1999, URL:http://www.ietf.org/rfc/rfc2616.txt

[RFC3261] "SIP: Session Initiation Protocol", J. Rosenberg et al., June 2002, URL: http://www.rfc-editor.org/rfc/rfc3261.txt

[RFC3966] "The tel URI for Telephone Numbers", H. Schulzrinne, December 2004, URL: http://www.ietf.org/rfc/rfc3966.txt

[RFC3986] "Uniform Resource Identifier (URI): Generic Syntax", R. Fielding et. al, January 2005, URL: http://www.ietf.org/rfc/rfc3986.txt

[RFC4627] "The application/json Media Type for JavaScript Object Notation (JSON)", D. Crockford, July 2006, URL: http://www.ietf.org/rfc/rfc4627.txt

[RFC4648] "The Base16, Base32, and Base64 Data Encodings", S. Josefsson, October 2006, URL: http://www.ietf.org/html/rfc4648

[SCRRULES] "SCR Rules and Procedures", Open Mobile Alliance™, OMA-ORG-SCR_Rules_and_Procedures, URL:http://www.openmobilealliance.org/

[W3C_URLENC] HTML 4.01 Specification, Section 17.13.4 Form content types, The World Wide Web Consortium, URL: http://www.w3.org/TR/html401/interact/forms.html#h-17.13.4.1

[W3C_XML11] W3C XML 1.1 Specification, URL: http://www.w3.org/TR/xml11/

[XMLSchema1] W3C Recommendation, XML Schema Part 1: Structures Second Edition, URL: http://www.w3.org/TR/xmlschema-1/

[XMLSchema2] W3C Recommendation, XML Schema Part 2: Data types Second Edition, URL: http://www.w3.org/TR/xmlschema-2/

2.2 Informative References

[OMADICT] "Dictionary for OMA Specifications", Version 2.9, Open Mobile Alliance™, OMA-ORG-Dictionary-V2_9, URL: http://www.openmobilealliance.org/

[ParlayREST_AudioCall] "RESTful bindings for Parlay X Web Services – Audio Call", Version 1.0, Open Mobile Alliance™, OMA-TS-ParlayREST-AudioCall-V1_0, URL: http://www.openmobilealliance.org/

[REST_WP] "Guidelines for RESTful Network APIs", Open Mobile Alliance™, OMA-WP-Guidelines_for_RESTful_Network_APIs, URL: http://www.openmobilealliance.org/

3. Terminology and Conventions

3.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope” and “Introduction”, are normative, unless they are explicitly indicated to be informative.

3.2 Definitions

For the purpose of this TS, all definitions from the OMA Dictionary apply [OMADICT].

3.3 Abbreviations

ACR	Anonymous Customer Reference
API	Application Programming Interface
ASR	Automatic Speech Recognition
HTTP	HyperText Transfer Protocol
JSON	JavaScript Object Notation
MIME	Multipurpose Internet Mail Extensions
OMA	Open Mobile Alliance
REST	REpresentational State Transfer
SCR	Static Conformance Requirements
SIP	Session Initiation Protocol
TS	Technical Specification
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
WP	White Paper
XML	eXtensible Markup Language
XSD	XML Schema Definition

4. Introduction

The Technical Specification for the RESTful Network API for Audio Call contains HTTP protocol bindings based on the Parlay X Audio Call Web Services [3GPP 29.199-11] specification, using the REST architectural style. The specification provides resource definitions, the HTTP verbs applicable for each of these resources, and the element data structures, as well as support material including flow diagrams and examples using the various supported message body formats (i.e. XML, JSON, and application/x-www-form-urlencoded).

4.1 Version 1.0

The RESTful Network API for Audio Call V1.0 is a republication of the ParlayREST AudioCall API V 1.0 [ParlayREST_AudioCall] as part of the suite of OMA RESTful Network APIs. Bug fixes and structural changes to fit that suite, but also functional changes have been applied.

Version 1.0 of the RESTful Network API for Audio Call keeps supporting the following operations:

- Playing a text message (e.g. using text-to-speech) to one or more call participants
- Playing an audio message to one or more call participants
- Playing a VoiceXML message to one or more call participants
- Playing a video message to one or more call participants
- Playing a message and collecting user keypresses on the phone keypad
- Playing a message and recording the user's voice

The following new functionality has been introduced:

- Using a script to engage the user in a dialog, and automatically recognize the user's spoken response
- Support for scope values used with authorization framework defined in [Autho4API_10]
- Support for Anonymous Customer Reference (ACR) as an end user identifier
- Support for "acr:auth" as a reserved keyword that identifies an end user

5. Audio Call API definition

This section is organized to support a comprehensive understanding of the Audio Call API design. It specifies the definition of all resources, definition of all data structures, and definitions of all operations permitted on the specified resources.

Common data types, naming conventions, fault definitions and namespaces are defined in [REST_NetAPI_Common].

The remainder of this document is structured as follows:

Section 5 starts with a diagram representing the resources hierarchy, followed by a table listing all the resources (and their URL) used by this API, along with the data structure and the supported HTTP verbs (section 5.1). What follows are the data structures (section 5.2). A sample of typical use cases is included in section 5.3, described as high level flow diagrams.

Section 6 contains the detailed specification for each of the resources. Each subsection defines the resource, the request URL variables that are common for all HTTP commands, the possible HTTP response codes, and the supported HTTP verbs. For each supported HTTP verb, a description of the functionality is provided, along with an example of a request and an example of a response. For each unsupported HTTP verb, the returned HTTP error status is specified, as well as what to return in the Allow header.

All examples in section 6 use XML as the format for the message body. Application/x-www-form-urlencoded examples are provided in Appendix C, while JSON examples are provided in Appendix D. Appendix B provides the Static Conformance Requirements (SCR).

Appendix E lists the Parlay X equivalent method for each supported REST resource and method combination, where applicable.

Appendix F provides a list of all light-weight resources, where applicable.

Appendix G defines authorization aspects to control access to the resources defined in this specification.

Note: Throughout this document client and application can be used interchangeably.

5.1 Resources Summary

This section summarizes all the resources used by the RESTful Network API for Audio Call.

The "apiVersion" URL variable SHALL have the value "v1" to indicate that the API corresponds to this version of the specification. See [REST_NetAPI_Common] which specifies the semantics of this variable.

The figure below visualizes the resource structure defined by this specification. Note that those nodes in the resource tree which have associated HTTP methods defined in this specification are depicted by solid boxes.

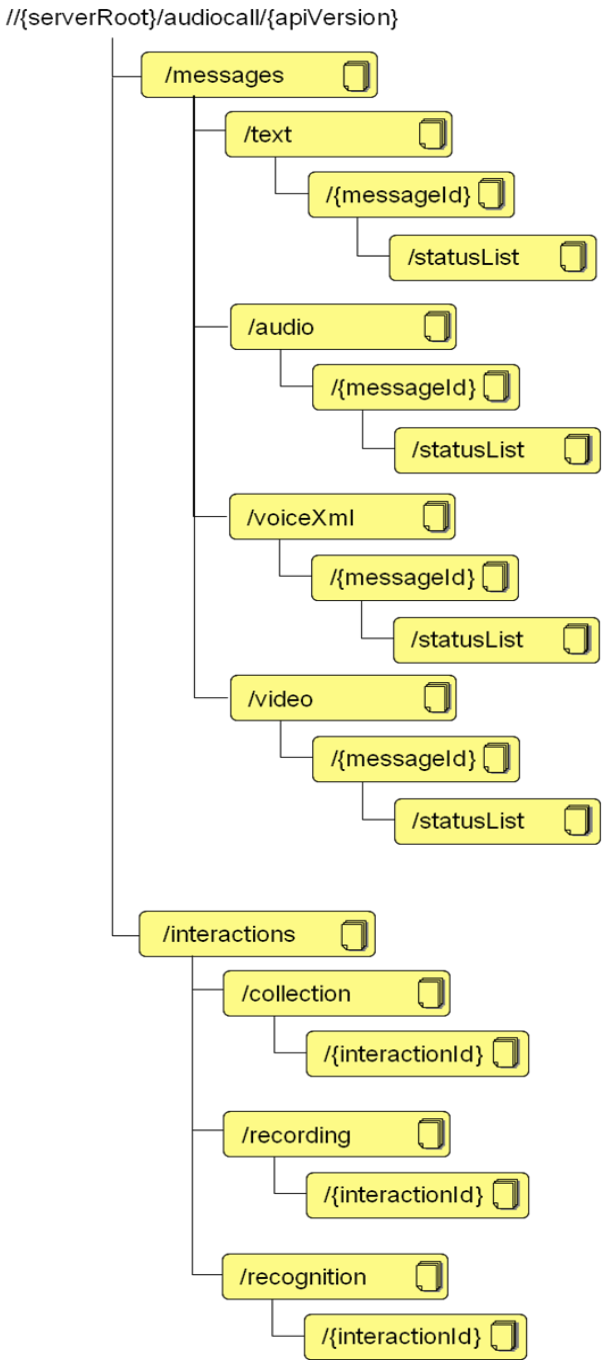

Figure 1 Resource structure defined by this specification

The following tables give a detailed overview of the resources defined in this specification, the data type of their representation and the allowed HTTP methods.

Purpose: To allow the client to manage audio, text, voiceXML and video messages

Resource	URL Base URL: http://{serverRoot}/audiocall/{apiVersion}	Data Structures	HTTP verbs			
			GET	PUT	POST	DELETE
All audio call messages	/messages	MessageList	Read all active audio call messages	no	no	no
Text messages	/messages/text	MessageList (used for GET) TextMessage (used for POST) common:ResourceReference (optional alternative for POST response)	Read all active text messages	no	Create new text message to be played to call participant(s)	no
Individual text message	/messages/text/{messageId}	TextMessage	Read text message	no	no	Terminate and remove text message
Individual text message status	/messages/text/{messageId}/statusList	MessageStatusList	Read message status	no	no	no
Audio messages	/messages/audio	MessageList (used for GET) AudioMessage (used for POST) common:ResourceReference (optional alternative for POST response)	Read all active audio messages	no	Create new audio message to be played to call participant(s)	no
Individual audio message	/messages/audio/{messageId}	AudioMessage	Read audio message	no	no	Terminate and remove audio message
Individual audio message status	/messages/audio/{messageId}/statusList	MessageStatusList	Read message status	no	no	no

Resource	URL Base URL: http://{serverRoot}/audiocall/{apiVersion}	Data Structures	HTTP verbs			
			GET	PUT	POST	DELETE
VoiceXML messages	/messages/voiceXml	MessageList (used for GET) VoiceXMLMessage (used for POST) common:ResourceReference (optional alternative for POST response)	Read all active VoiceXML messages	no	Create new VoiceXML message to be played to call participant(s)	no
Individual VoiceXML message	/messages/voiceXml/{messageId}	VoiceXMLMessage	Read VoiceXML message	no	no	Terminate and remove audio message
Individual VoiceXML message status	/messages/voiceXml/{messageId}/statusList	MessageStatusList	Read message status	no	no	no
Video messages	/messages/video	MessageList (used for GET) VideoMessage (used for POST) common:ResourceReference (optional alternative for POST response)	Read all active video messages	no	Create new video message to be played to call participant(s)	no
Individual video message	/messages/video/{messageId}	VideoMessage	Read video message	no	no	Terminate and remove audio message
Individual video message status	/messages/video/{messageId}/statusList	MessageStatusList	Read message status	no	no	no

Purpose : To allow the client to manage media capture and interactions

Resource	**URL Base URL: http://{serverRoot}/audiocall/{apiVersion}**	Data Structures	HTTP verbs			
			GET	PUT	POST	DELETE
All media capture interactions	/interactions	InteractionList	Read all active media capture interactions	no	no	no
Play-and-collect interactions	/interactions/collection	InteractionList (used for GET) DigitCapture (used for POST) common:ResourceReference (optional alternative for POST response)	Read all active audio calls where media is being played to participant(s) and digits being collected	no	Play a media file call participant(s) and and collect digits from the participant(s)	no
Individual play-and-collect interaction	/interactions/collection/{interactionId}	DigitCapture	Read individual play-and-collect interaction	no	no	Stop interaction and remove information
Play media and record participant(s) response	/interactions/recording	InteractionList (used for GET) RecordingCapture (used for POST) common:ResourceReference (optional alternative for POST response)	Read all active audio calls where media is being played to participant(s) and information being recorded from the partcipant(s)	no	Play a media file to call participant(s) and and record information (media) from the participant(s)	no
Individual play-and-record interaction	/interactions/recording/{interactionId}	RecordingCapture	Read individual play-and-record interaction	no	no	Stop interaction and remove information
Engage the participant(s) in a dialog and recognize their spoken response	/interactions/recognition	InteractionList (used for GET) RecognitionCapture (used for POST) common:ResourceReference (optional alternative for POST response)	Read all active audio calls where a script is used to engage the participant(s) in a dialog and recognize their spoken responses	no	Engage the participant(s) in a dialog and recognize their spoken response	no

Resource	**URL** **Base URL: http://{serverRoot}/audiocall/{apiVersion}**	Data Structures	HTTP verbs			
			GET	PUT	POST	DELETE
Individual recognition	/interactions/recognition/{interactionId}	RecognitionCapture	Read individual recognition interaction	no	no	Stop interaction and remove information

5.2 Data Types

5.2.1 XML Namespaces

The namespace for the Audio Call data types is:

urn:oma:xml:rest:netapi:audiocall:1

The 'xsd' namespace is used in the present document to refer to the XML Schema data types defined in XML Schema [XMLSchema1, XMLSchema2]. The 'common' namespace is used in the present document to refer to the data types defined in [REST_NetAPI_Common]. The use of the names 'xsd' and ‘common’ is not semantically significant.

The XML schema for the data structures defined in the section below is given in [REST_SUP_AudioCall].

Applications following the RESTful Network API for AudioCall V 1.0 specification SHALL use the namespace urn:oma:xml:rest:netapi:audiocall:1.

Note: Server implementations can choose to also support the legacy namespace urn:oma:xml:rest:audiocall:1 for the AudioCall data types, in order to allow backwards-compatibility with [ParlayREST_AudioCall] applications. Use of this legacy namespace is deprecated and support is foreseen to be withdrawn in future versions of this specification. In messages sent from the server to the application, the legacy namespace is suggested to be used by the server if it was used by a legacy application in the corresponding request or subscription message.

5.2.2 Structures

The subsections of this section define the data structures used in the Audio Call API.

Some of the structures can be instantiated as so-called root elements.

For structures that contain elements which describe a user identifier, the statements in section 6 regarding 'tel', 'sip' and 'acr' URI schemes apply.

5.2.2.1 Type: DigitConfig

Defines the configuration parameters for the input part of capture (collection) of digits from the phone keypad.

Name	Type	Optional	Description
maxDigits	xsd:unsignedInt	Yes	The maximum number of digits that will be collected. If not given, the behaviour is implementation-specific.
minDigits	xsd:unsignedInt	Yes	The minimum number of digits that will be collected. If this isn't achieved, then a default prompt shall be played requesting for more digits to be entered. If not given, the behaviour is implementation-specific.
endChar	xsd:string	Yes	If the user enters via the phone keypad the character specified in this element, this indicates to stop collecting digits. This string SHALL have a minimum and maximum length of one; valid characters are the digits ’0’ to ‘9’, asterisk (‘*’) and hash (‘#’). If pressed, the end character SHALL neither be returned nor count towards the number of digits to be collected as defined by minDigits and maxDigits.

			Note: Without this the user will have to wait for a timeout after entering his/her digits. Note: In case endChar is not given, the behavior (i.e. whether or not a particular character such as '#' triggers the end of the digit collection) is network-specific.

5.2.2.2 Type: RecConfig

Defines the configuration parameters for the input part of the recording of a user's voice.

Name	Type	Optional	Description
recFileLocation	xsd:anyURI	Yes	The location for storing the information recorded from the terminal. If not given, the behaviour is implementation-specific.
maxRecordingLength	xsd:int	Yes	The maximum time to record the media for. Unit is seconds.
endChar	xsd:string	Yes	If the user enters via the phone keypad the character specified in this element, this indicates to stop recording. This string SHALL have a minimum and maximum length of one; valid characters are the digits '0' to '9', asterisk ('*') and hash ('#'). Note: Without this the user will have to wait for a timeout after entering his/her digits. Note: In case endChar is not given, the behavior (i.e. whether or not a particular character such as '#' triggers the end of the recording) is network-specific.

5.2.2.3 Type: ASRConfig

Defines the configuration parameters for the automatic speech recognition engine.

Name	Type	Optional	Description
scriptFileLocation	xsd:anyURI	Choice	The location of the script file, e.g. VoiceXML. The meaning of this URI is implementation specific. Note that usually, announcement media are pre-loaded into specific network nodes, but some implementations might also allow playing content available on internet servers.
script	xsd:any	Choice	The script embedded in the configuration parameter set. The format of the script is out of scope of this

			specification. If the ASRConfig data structure is represented in the XML format, this element can carry XML encoded or base64 [RFC4648] encoded scripts. If the ASRConfig data structure is represented in the JSON format, this element MUST be base64 [RFC4648] encoded.
scriptType	xsd:string	No	The MIME type of the script.
scriptEncoding	xsd:string	Yes	If present and set to "base64", the script is base64-encoded.
endChar	xsd:string	Yes	If the user enters via the phone keypad the character specified in this element, this indicates to stop the dialog. This string SHALL have a minimum and maximum length of one; valid characters are the digits '0' to '9', asterisk ('*') and hash ('#'). Note: Without this the user will have to wait for a timeout after entering his/her digits. Note: In case endChar is not given, the behavior (i.e. whether or not a particular character such as '#' triggers the end of the recording) is network-specific.

XSD modelling uses a "choice" to select either "scriptFileLocation" or "script".

5.2.2.4 Type: PlayConfig

Defines the configuration parameters for the playback of the prompt as part of the recording of a user's voice.

Note that if the file to be played is of format VoiceXML, this may include interactions on its own. These interactions are processed internally in the VoiceXML script and are not returned to the Application.

Name	Type	Optional	Description
playFileLocation	xsd:anyURI	Choice	The location of the file that will be played to the endpoint, including VoiceXML script location. The meaning of this URI is implementation specific. Note that usually, announcement media are pre-loaded into specific network nodes, but some implementations might also allow playing content available on internet servers.
textString	xsd:string	Choice	The text to be converted by a Text-To-Speech engine.
messageFormat	AnnouncementFormat	No	The type of announcement prompt to play to the end user.
mediaType	xsd:string	Yes	MIME media type of the content to be played.
interruptMedia	xsd:boolean	No	Indicates whether the application allows the end user to

			interrupt, or pause, the prompt.

XSD modelling uses a “choice” to select either a playFileLocation or a textString to refer to the message to be played.

5.2.2.5 Type: MessageStatusList

Status of a particular message for a list of participants.

Name	Type	Optional	Description
messageStatus	MediaMessageStatus[0..unbounded]	Yes	Message status.
resourceURL	xsd:anyURI	No	Self referring URL.

A root element named messageStatusList of type MessageStatusList is allowed in response bodies.

5.2.2.6 Type: MediaMessageStatus

Status of the message for each callParticipant after message operation has been invoked.

Name	Type	Optional	Description
callParticipant	xsd:anyURI	No	Call Participant identifier (e.g. 'sip' URI, 'tel' URI, 'acr' URI).
status	MessageStatus	No	Current playing status of the participant.

5.2.2.7 Type: MessageList

List of messages.

Name	Type	Optional	Description
textMessage	TextMessage[0..unbounded]	Yes	List of text messages.
audioMessage	AudioMessage[0..unbounded]	Yes	List of audio messages.
voiceXmlMessage	VoiceXMLMessage[0..unbounded]	Yes	List of VoiceXML messages.
videoMessage	VideoMessage[0..unbounded]	Yes	List of video messages.
resourceURL	xsd:anyURI	No	Self referring URL.

A root element named messageList of type MessageList is allowed inresponse bodies.

5.2.2.8 Type: TextMessage

This structure represents a text message, usually presented via Text-to-Speech.

Name	Type	Optional	Description
callSessionIdentifier	xsd:string	Choice	Identifies the call session to which the message is played.

link	common:Link	Choice	Identifie (by a REST resource URL) the call session to which the message is played. The “rel” attribute MUST be equal to “CallSessionInformation”, and the link MUST point to a resource of that type.
callParticipant	xsd:anyURI [0..unbounded]	Yes	The set of participant addresses (e.g. 'sip' URI, 'tel' URI, 'acr' URI) contained within the callSession to which the message is to be played. If no participants are specified, the message is played to all participants.
text	common:languageString	No	Text to process with a Text-To-Speech engine. The language of the text SHOULD be defined by populating the attribute xml:lang of this element.
messageStatusList	MessageStatusList	Yes	Message status list. Added by the server after creating the resource, and kept up-to-date by the server to reflect the actual status changes.
charging	common:ChargingInformation	Yes	Charge to apply for the playing of this message. If charging is not supported then a PolicyException (POL0008) will be returned.
clientCorrelator	xsd:string	Yes	A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server. This element SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids playing the same message twice in such situations. In case the element is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it.
resourceURL	xsd:anyURI	Yes	Self referring URL. The resourceURL SHALL NOT be included in POST requests by the client, but MUST be included in POST requests representing notifications by the server to the client, when a complete representation of the resource is embedded in the notification. The resourceURL MUST also be included in responses to any HTTP method that returns an entity body, and in PUT requests.

A root element named textMessage of type TextMessage is allowed in request and response bodies.

Note that the clientCorrelator is used for purposes of error recovery as specified in [REST_NetAPI_Common], and internal client purposes. The server is NOT REQUIRED to use the clientCorrelator value in any form in the creation of the URL of the resource. The document [REST_NetAPI_Common] provides a recommendation regarding the generation of the value of this field.

XSD modelling uses a “choice” to select either callSessionIdentifier or link to refer to a call session.

5.2.2.9 Type: MediaMessage

This structure represents a generic media message.

Name	Type	Optional	Description
callSessionIdentifier	xsd:string	Choice	Identifies the call session to which the message is played.
link	common:Link	Choice	Identifies (by a REST resource URL) the call session to which the message is played. The “rel” attribute MUST be equal to “CallSessionInformation”, and the link MUST point to a resource of that type.
callParticipant	xsd:anyURI [0..unbounded]	Yes	The set of participant addresses (e.g. 'sip' URI, 'tel' URI, 'acr' URI) contained within the callSession to which the message is to be played. If no participants are specified, the message is played to all participants.
mediaUrl	xsd:anyURI	No	Location of content (audio, video, voiceXml) to play. The meaning of this URI is implementation specific. Note that usually, announcement media are pre-loaded into specific network nodes, but some implementations might also allow playing content available on internet servers.
mediaType	xsd:string	Yes	MIME media type of the content to be played.
messageStatusList	MessageStatusList	Yes	Message status list. Added by the server after creating the resource, and kept up-to-date by the server to reflect the actual status changes.
charging	common:ChargingInformation	Yes	Charge to apply for the playing of this message. If charging is not supported then a PolicyException (POL0008) will be returned.

clientCorrelator	xsd:string	Yes	A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server. This element SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids playing the same message twice in such situations. In case the element is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it.
resourceURL	xsd:anyURI	Yes	Self referring URL. The resourceURL SHALL NOT be included in POST requests by the client, but MUST be included in POST requests representing notifications by the server to the client, when a complete representation of the resource is embedded in the notification. The resourceURL MUST also be included in responses to any HTTP method that returns an entity body, and in PUT requests.

Regarding the clientCorrelator field, the note in section 5.2.2.8 applies.

XSD modelling uses a “choice” to select either callSessionIdentifier or link to refer to a call session.

5.2.2.10 Type: AudioMessage

This structure represents a media message in audio format.

It inherits all fields of the type MediaMessage (see section 5.2.2.9). There are no fields added in this version of the specification.

A root element named audioMessage of type AudioMessage is allowed in request and response bodies.

5.2.2.11 Type: VideoMessage

This structure represents a media message in video format.

It inherits all fields of the type MediaMessage (see section 5.2.2.9). There are no fields added in this version of the specification.

A root element named videoMessage of type VideoMessage is allowed in request and response bodies.

5.2.2.12 Type: VoiceXMLMessage

This structure represents a media message in VoiceXML format.

It inherits all fields of the type MediaMessage (see section 5.2.2.9). There are no fields added in this version of the specification.

A root element named voiceXMLMessage of type VoiceXMLMessage is allowed in request and response bodies.

5.2.2.13 Type: InteractionList

List of interactions.

Name	Type	Optional	Description
digitCapture	DigitCapture[0..unbounded]	Yes	List of digit capture interactions.
recordingCapture	RecordingCapture[0..unbounded]	Yes	List of recording capture interactions.
recognitionCapture	RecognitionCapture[0..unbounded]	Yes	List of automatic speech recognition interactions.
resourceURL	xsd:anyURI	No	Self referring URL.

A root element named interactionList of type InteractionList is allowed in response bodies.

5.2.2.14 Type: DigitCapture

This structure represents a digit capture interaction, which combines the playback of a message with the capturing of key presses on the phone keyboard using DTMF.

Name	Type	Optional	Description
callSessionIdentifier	xsd:string	Choice	Identifies the call session for the media interaction.
link	common:Link	Choice	Identifies (by a REST resource URL) the call session for the media interaction. The “rel” attribute MUST be equal to “CallSessionInformation”, and the link MUST point to a resource of that type.
callParticipant	xsd:anyURI [0..unbounded]	Yes	The set of participant addresses (e.g. 'sip' URI, 'tel' URI, 'acr' URI) contained within the call session to which the message is to be played and from which the digits are to be captured. If no participants are specified, this applies to all participants in the call session.
playingConfiguration	PlayConfig	No	Configuration parameters related to the playing of a media file.
digitConfiguration	DigitConfig	No	Configuration parameters related to digit collection.

clientCorrelator	xsd:string	Yes	A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server. This element SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids starting the digit capture twice in such situations. In case the element is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it.
resourceURL	xsd:anyURI	Yes	Self referring URL. The resourceURL SHALL NOT be included in POST requests by the client, but MUST be included in POST requests representing notifications by the server to the client, when a complete representation of the resource is embedded in the notification. The resourceURL MUST also be included in responses to any HTTP method that returns an entity body, and in PUT requests.

A root element named digitCapture of type DigitCapture is allowed in request and response bodies.

Regarding the clientCorrelator field, the note in section 5.2.2.8 applies.

XSD modelling uses a “choice” to select either callSessionIdentifier or link to refer to a call session.

5.2.2.15 Type: RecordingCapture

This structure represents a recording interaction, which combines the playback of a message with recording the voice of a participant.

Name	Type	Optional	Description
callSessionIdentifier	xsd:string	Choice	Identifies the call session for the media interaction.
link	common:Link	Choice	Identifies (by a REST resource URL) the call session for the media interaction. The “rel” attribute MUST be equal to “CallSessionInformation”, and the link MUST point to a resource of that type.
callParticipant	xsd:anyURI [0..unbounded]	Yes	The set of participant addresses (e.g. 'sip' URI, 'tel' URI, 'acr' URI) contained within the call session to which the message is to be played and from which the recordings are to be captured. If no participants are specified, this applies to all participants in the call session.

playingConfiguration	PlayConfig	No	Configuration parameters related to the playing of a media file.
recordingConfiguration	RecConfig	No	Configuration parameters related to media recording.
clientCorrelator	xsd:string	Yes	A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server. This element SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids starting the recording twice in such situations. In case the element is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it.
resourceURL	xsd:anyURI	Yes	Self referring URL. The resourceURL SHALL NOT be included in POST requests by the client, but MUST be included in POST requests representing notifications by the server to the client, when a complete representation of the resource is embedded in the notification. The resourceURL MUST also be included in responses to any HTTP method that returns an entity body, and in PUT requests.

A root element named recordingCapture of type RecordingCapture is allowed in request and response bodies.

Regarding the clientCorrelator field, the note in section 5.2.2.8 applies.

XSD modelling uses a “choice” to select either callSessionIdentifier or link to refer to a call session.

5.2.2.16 Type: RecognitionCapture

This structure represents a automatic speech recognition (ASR) interaction, which realizes a dialog as defined by a script file, and converts the user’s spoken responses into simple or structured text.

Name	Type	Optional	Description
callSessionIdentifier	xsd:string	Choice	Identifies the call session for the media interaction.
link	common:Link	Choice	Identifies (by a REST resource URL) the call session for the media interaction. The “rel” attribute MUST be equal to “CallSessionInformation”, and the link MUST point to a resource of that type.

callParticipant	xsd:anyURI [0..unbounded]	Yes	The set of participant addresses (e.g. 'sip' URI, 'tel' URI, 'acr' URI) contained within the call session with whom to interact. If no participants are specified, this applies to all participants in the call session.
asrConfiguration	ASRConfig	No	Configuration parameters for the automatic speech recognizer
clientCorrelator	xsd:string	Yes	A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server. This element SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids starting the speech recognition dialog twice in such situations. In case the element is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it.
resourceURL	xsd:anyURI	Yes	Self referring URL. The resourceURL SHALL NOT be included in POST requests by the client, but MUST be included in POST requests representing notifications by the server to the client, when a complete representation of the resource is embedded in the notification. The resourceURL MUST also be included in responses to any HTTP method that returns an entity body, and in PUT requests.

A root element named recognitionCapture of type recognitionCapture is allowed in request and response bodies.

Regarding the clientCorrelator field, the note in section 5.2.2.8 applies.

XSD modelling uses a “choice” to select either callSessionIdentifier or link to refer to a call session.

5.2.3 Enumerations

The subsections of this section define the enumerations used in the Audio Call API.

5.2.3.1 Enumeration: MessageStatus

Status of the message after play message operation has been invoked. Final states are Played, Error and Terminated.

Enumeration	Description
Played	Message has been played

Playing	Message is currently playing
Pending	Message has not yet started playing
Error	An error has occurred, message will not be played
Terminated	The message was terminated by a request from the application. Note that this state value is not available in Parlay X.

5.2.3.2 Enumeration: AnnouncementFormat

This enumeration defines values representing the different formats of an announcement (prompt) in an interaction.

Enumeration	Description
Audio	Announcement is in Audio format
VoiceXML	Announcement is in VoiceXML format
TextToSpeech	Announcement is in TextToSpeech format
Video	Announcement is in Video format
ApplicationSpecificFormat	Announcement is in an ApplicationSpecificFormat

5.2.4 Values of the Link "rel" attribute

The "rel" attribute of the Link element is a free string set by the server implementation, to indicate a relationship between the current resource and an external resource. The following are possible strings (list is non-exhaustive, and can be extended):

- MessageStatusList
- MessageList
- TextMessage
- AudioMessage
- VideoMessage
- VoiceXMLMessage
- InteractionList
- DigitCapture
- RecordingCapture
- RecognitionCapture

These values indicate the kind of resource that the link points to.

5.3 Sequence Diagrams

The following subsections describe the resources, methods and steps involved in typical scenarios.

5.3.1 Play Audio Message and Check Status

This figure below shows a scenario for initiating the playing of an audio message to a call participant and then checking the status as the message is played.

The resources:

- To initiate playing of the audio message, create a new resource under

  http://{serverRoot}/audiocall/{apiVersion}/messages/audio

- To get the status of the message, do either a or b:

  a. read the newly created resource including the status of the message
  http://{serverRoot}/audiocall/{apiVersion}/messages/audio/{messageId}

  b. directly read the status of the message
  http://{serverRoot}/audiocall/{apiVersion}/messages/audio/{messageId}/statusList

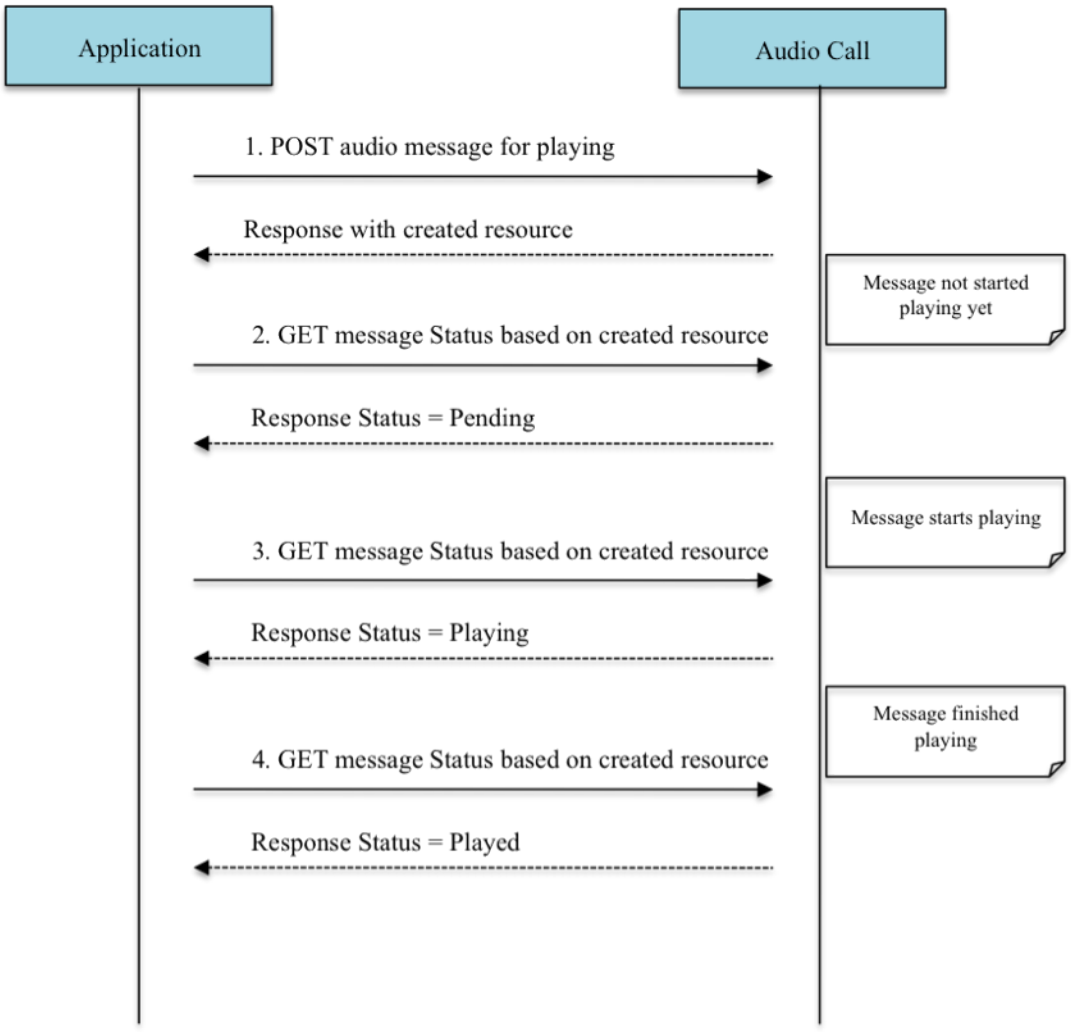

Figure 2 Play audio message and check status

Outline of the flows:

1. An application initiates the playing of an audio message to a set of or all participants in a call using POST and receives the created request resource with a resource URL containing the messageId.
2. The application requests the status of playing the, and receives “Pending” for some participants as the message did not start playing
3. The application requests the status of playing the message, and receives “Playing” for some participants as the message is actually playing
4. The application requests the status of playing the message, and receives “Played” for some participants as the message has finished playing

For steps 2-4, the application achieves this either by

a) The application requests the resource of the message using GET with the given resource URL (containing the messageId) and receives a representation of the resource which includes the status information, or

b) The application *directly* requests the status information, using GET with a specific child “statusList” of the given resource URL (containing the messageId).

Note that for each participant, a different status could be returned in steps 2-4.

5.3.2 Play Audio Message and Terminate the Playing of the Message

This figure below shows a scenario for initiating the playing of an audio message to a call participant and then terminating playing of the message.

The resources:

- To initiate playing of the audio message, create new a resource under **http://{serverRoot}/audiocall/{apiVersion}/messages/audio**
- To terminate the message, delete the newly created resource **http://{serverRoot}/audiocall/{apiVersion}/messages/audio/{messageId}**

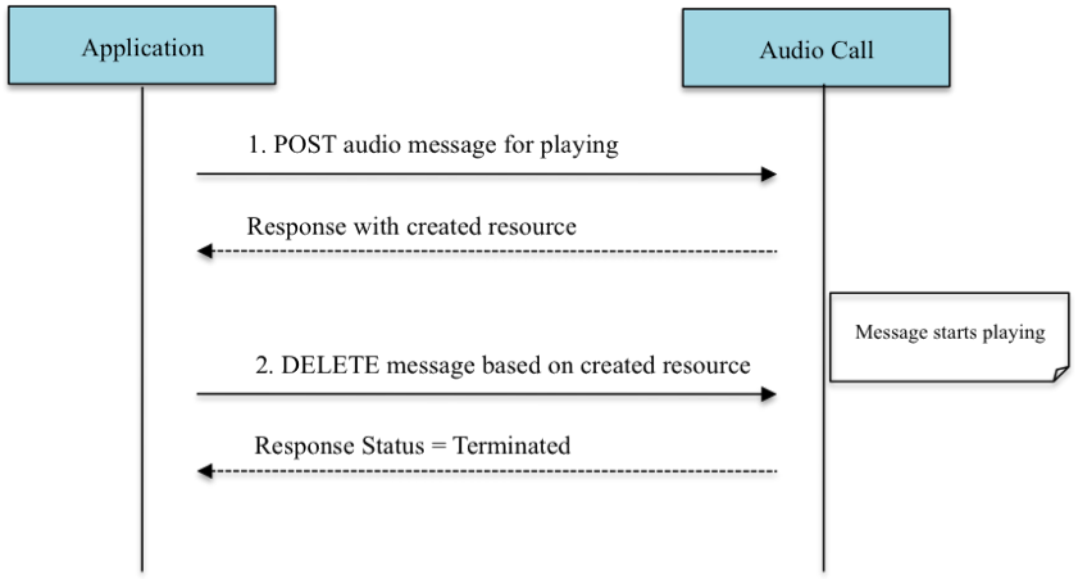

Figure 3 Play audio message and terminate the playing of the message

Outline of the flows:

1. An application initiates the playing of an audio message to a set of or all participants in a call using POST and receives the created request resource with a resource URL containing the messageId.
2. The application terminates playing of the message with the given resource URL (containing the messageId) using DELETE on that resourceURL. It receives the status of the message for each participant as part of the representation of the resource delivered in the response of the DELETE operation.

5.3.3 Play Media and Collect Digits

This figure below shows a scenario for initiating the playing of an announcement to a call participant, capturing digits entered by the participant, and then notifying the application about the digits entered. Note that the Audio Call service also allows the recording of a voice response from the call participant, instead of collecting digits.

This is an asynchronous process involving a notification of the application, which is carried out by the Call Notification service, not the Audio Call service.

The resources:

- The resources for Call Notification are defined in[REST-NetAPI_CallNotif]
- To play a media file to a set of or all participants and collect input from the users via the phone keypad, create a new resource under
  http://{serverRoot}/audiocall/{apiVersion}/interactions/collection

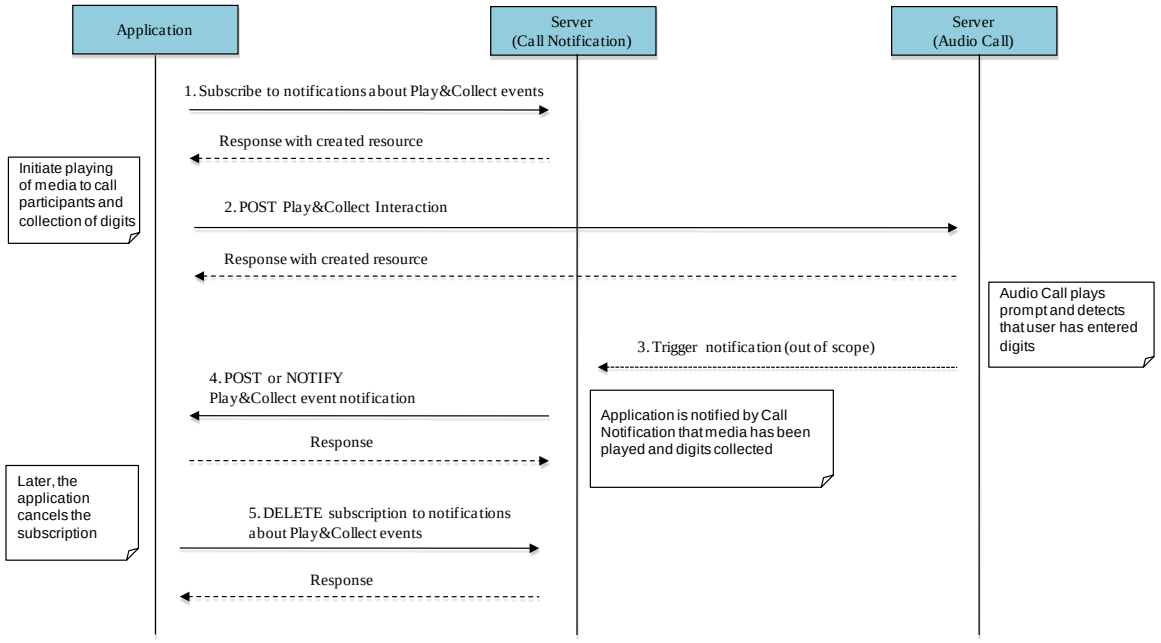

Figure 4 Play media and collect digits

Outline of the flows:

1. An application subscribes to notifications about play-and-collect events for an ongoing call using POST.
2. The application requests the playing of a media file to a set of or all call participants and the collection of their inputs using POST and receives the created request resource with a resource URL containing the interactionId.
3. Once the Audio Call service has finished the playback of the media and the collection of the inputs, it triggers the Call Notification service [REST_TS_ CallNotif] by means out of scope of this specification. The Call Notification service sends a play-and-collect notification as defined in [REST_TS_ CallNotif] .

4. The application terminates the subscription to notifications about play-and-collect events for the ongoing call using DELETE.

6. Detailed specification of the resources

The following applies to all resources defined in this specification regardless of the representation format (i.e. XML, JSON, application/x-www-form-urlencoded):

- Reserved characters in URL variables (parts of a URL denoted below by a name in curly brackets) MUST be percent-encoded according to [RFC3986]. Note that this always applies, no matter whether the URL is used as a Request URL or inside the representation of a resource (such as in "resourceURL" and "link" elements).

- If a user identifier (e.g. address, callParticipantId, etc.) of type anyURI is in the form of an MSISDN, it MUST be defined as a global number according to [RFC3966] (e.g. tel:+19585550100). The use of characters other than digits and the leading "+" sign SHOULD be avoided in order to ensure uniqueness of the resource URL. This applies regardless of whether the user identifier appears in a URL variable or in a parameter in the body of an HTTP message.

- If a user identifier (e.g. address, callParticipantId, etc.) of type anyURI is in the form of a SIP URI, it MUST be defined according to [RFC3261].

- If a user identifier (e.g. address, callParticipantId, etc.) of type anyURI is in the form of an Anonymous Customer Reference (ACR), it MUST be defined according to [IETF_ACR_draft], i.e. it MUST include the protocol prefix 'acr:' followed by the ACR.

    - The ACR 'auth' is a supported reserved keyword, and MUST NOT be assigned as an ACR to any particular end user. See G.1.2 for details regarding the use of this reserved keyword.

- For requests and responses that have a body, the following applies: in the requests received, the server SHALL support JSON and XML encoding of the parameters in the body, and MAY support application/x-www-form-urlencoded parameters in the body. The Server SHALL return either JSON or XML encoded parameters in the response body, according to the result of the content type negotiation as specified in [REST_NetAPI_Common]. In notifications to the Client, the server SHALL use either XML or JSON encoding, depending on which format the client has specified in the related subscription. The generation and handling of the JSON representations SHALL follow the rules for JSON encoding in HTTP Requests/Responses as specified in [REST_ NetAPI_Common].

6.1 Resource: All audio call messages

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages

This resource is used to obtain all active audio call messages.

6.1.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.1.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.1.3 GET

This operation is used for reading a list of all active audio call messages.

6.1.3.1 Example: Retrieving a list of all active audio call messages (Informative)

6.1.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages HTTP/1.1
Accept: application/xml
Host: example.com
```

6.1.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:messageList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <textMessage>
      <callSessionIdentifier>A45678</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <callParticipant>tel:+19585550102</callParticipant>
      <text xml:lang="en">Welcome to the telephone conference</text>
      <messageStatusList>
         <messageStatus>
            <callParticipant>tel:+19585550101</callParticipant>
            <status>Played</status>
         </messageStatus>
         <messageStatus>
            <callParticipant>tel:+19585550102</callParticipant>
            <status>Pending</status>
         </messageStatus>
         <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList</resourceURL>
      </messageStatusList>
      <clientCorrelator>12345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123</resourceURL>
   </textMessage>
  <audioMessage>
     <callSessionIdentifier>B45678</callSessionIdentifier>
     <callParticipant>tel:+19585550101</callParticipant>
     <callParticipant>tel:+19585550102</callParticipant>
      <mediaUrl>http://www.example.com/ann1.mp3</mediaUrl>
            <mediaType>audio/mpeg</mediaType>
       <messageStatusList>
         <messageStatus>
            <callParticipant>tel:+19585550101</callParticipant>
```

```
            <status>Played</status>
          </messageStatus>
          <messageStatus>
            <callParticipant>tel:+19585550102</callParticipant>
            <status>Pending</status>
          </messageStatus>
          <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList</resourceURL>
      </messageStatusList>
      <clientCorrelator>22345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123</resourceURL>
  </audioMessage>
  <audioMessage>
      <callSessionIdentifier>C45678</callSessionIdentifier>
      <callParticipant>tel:+19585550103</callParticipant>
      <callParticipant>tel:+19585550104</callParticipant>
      <mediaUrl>http://www.example.com/ann2.mp3</mediaUrl>
      <mediaType>audio/mpeg</mediaType>
      <messageStatusList>
          <messageStatus>
            <callParticipant>tel:+19585550103</callParticipant>
            <status>Played</status>
          </messageStatus>
          <messageStatus>
            <callParticipant>tel:+19585550104</callParticipant>
            <status>Playing</status>
          </messageStatus>
          <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg456/statusList</resourceURL>
      </messageStatusList>
      <clientCorrelator>32345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg456</resourceURL>
  </audioMessage>
  <voiceXmlMessage>
      <callSessionIdentifier>D45678</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <callParticipant>tel:+19585550102</callParticipant>
      <mediaUrl>http://www.example.com/ann1.vxml</mediaUrl>
       <messageStatusList>
          <messageStatus>
            <callParticipant>tel:+19585550101</callParticipant>
            <status>Played</status>
          </messageStatus>
          <messageStatus>
            <callParticipant>tel:+19585550102</callParticipant>
            <status>Pending</status>
          </messageStatus>
          <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList</resourceURL>
      </messageStatusList>
      <clientCorrelator>42345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123</resourceURL>
  </voiceXmlMessage >
  <videoMessage>
      <callSessionIdentifier>E45678</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <callParticipant>tel:+19585550102</callParticipant>
      <mediaUrl>http://www.example.com/ann1.mp4</mediaUrl>
      <mediaType>video/mp4</mediaType>
```



```
      <messageStatusList>
        <messageStatus>
          <callParticipant>tel:+19585550101</callParticipant>
          <status>Played</status>
        </messageStatus>
        <messageStatus>
          <callParticipant>tel:+19585550102</callParticipant>
          <status>Pending</status>
        </messageStatus>
        <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList</resourceURL>
</messageStatusList>
     <clientCorrelator>52345</clientCorrelator>
     <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123</resourceURL>
  </videoMessage >
  <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages</resourceURL>
</ac:messageList>
```

6.1.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.1.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.1.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.2 Resource: Text messages

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages/text

This resource is used to create an audio call using a text message as the media, and to retrieve a list of all active audio call text messages.

6.2.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.2.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.2.3 GET

This operation is used for reading a list of all active audio call text messages.

6.2.3.1 Example: Retrieving a list of all active audio call text messages (Informative)

6.2.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages/text HTTP/1.1
Accept: application/xml
Host: example.com
```

6.2.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:messageList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
  <textMessage>
    <callSessionIdentifier>A45678</callSessionIdentifier>
    <callParticipant>tel:+19585550101</callParticipant>
    <callParticipant>tel:+19585550102</callParticipant>
    <text xml:lang="en"> Welcome to the telephone conference</text>
    <messageStatusList>
      <messageStatus>
        <callParticipant>tel:+19585550101</callParticipant>
        <status>Played</status>
      </messageStatus>
      <messageStatus>
        <callParticipant>tel:+19585550102</callParticipant>
        <status>Pending</status>
      </messageStatus>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList</resourceURL>
    </messageStatusList>
    <clientCorrelator>12345</clientCorrelator>
    <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123</resourceURL>
  </textMessage>
  <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text</resourceURL>
</ac:messageList>
```

6.2.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the ‘Allow: GET, POST’ field in the response as per section 14.7 of [RFC2616].

6.2.5 POST

This operation is used to create a new text message to be played to call participants.

6.2.5.1 Example 1: Creating an audio call text message, response with copy of created resource (Informative)

6.2.5.1.1 Request

```
POST /exampleAPI/audiocall/v1/messages/text HTTP/1.1
Accept: application/xml
Content-Length: nnnn
Content-Type: application/xml
Host: example.com

<?xml version="1.0" encoding="UTF-8"?>
<ac:textMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>A45678</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <callParticipant>tel:+19585550102</callParticipant>
   <text xml:lang="en">Welcome to the telephone conference</text>
   <clientCorrelator>12345</clientCorrelator>
</ac:textMessage>
```

6.2.5.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/messages/text/msg123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:textMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
      <callSessionIdentifier>A45678</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <callParticipant>tel:+19585550102</callParticipant>
      <text xml:lang="en">Welcome to the telephone conference</text>
      <messageStatusList>
         <messageStatus>
            <callParticipant>tel:+19585550101</callParticipant>
            <status>Pending</status>
         </messageStatus>
         <messageStatus>
            <callParticipant>tel:+19585550102</callParticipant>
            <status>Pending</status>
         </messageStatus>
         <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList</resourceURL>
      </messageStatusList>
      <clientCorrelator>12345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123</resourceURL>
   </ac:textMessage>
```


6.2.5.2 Example 2: Creating an audio call text message, response with location of created resource (Informative)

6.2.5.2.1 Request

```
POST /exampleAPI/audiocall/v1/messages/text HTTP/1.1
Accept: application/xml
Content-Length: nnnn
Content-Type: application/xml
Host: example.com

<?xml version="1.0" encoding="UTF-8"?>
<ac:textMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>A45678</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <callParticipant>tel:+19585550102</callParticipant>
   <text xml:lang="en">Welcome to the telephone conference</text>
   <clientCorrelator>12345</clientCorrelator>
</ac:textMessage>
```

6.2.5.2.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/messages/text/msg123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<common:resourceReference xmlns:common="urn:oma:xml:rest:netapi:common:1">
  <resourceURL> http://example.com/exampleAPI/audiocall/v1/messages/text/msg123</resourceURL>
</common:resourceReference>
```

6.2.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, POST' field in the response as per section 14.7 of [RFC2616].

6.3 Resource: Individual text message

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages/text/{messageId}

This resource is used to retrieve or terminate an individual active audio call text message.

6.3.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description

serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.
messageId	identifier of the message resource.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.3.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.3.3 GET

This operation is used to retrieve an individual active audio call text message.

6.3.3.1 Example: Retrieving an active audio call text message (Informative)

This example shows also an alternative way to indicate desired content type in response from the server, by using URL query parameter "?resFormat" which is described in [REST_NetAPI_Common].

6.3.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages/text/msg123?resFormat=XML HTTP/1.1
Host: example.com
```

6.3.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:textMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
      <callSessionIdentifier>A45678</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <callParticipant>tel:+19585550102</callParticipant>
      <text xml:lang="en">Welcome to the telephone conference</text>
      <messageStatusList>
         <messageStatus>
            <callParticipant>tel:+19585550101</callParticipant>
            <status>Played</status>
         </messageStatus>
         <messageStatus>
            <callParticipant>tel:+19585550102</callParticipant>
            <status>Pending</status>
         </messageStatus>
         <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList</resourceURL>
      </messageStatusList>
```



```
    <clientCorrelator>12345</clientCorrelator>
    <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123</resourceURL>
  </ac:textMessage>
```

6.3.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, DELETE' field in the response as per section 14.7 of [RFC2616].

6.3.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, DELETE' field in the response as per section 14.7 of [RFC2616].

6.3.6 DELETE

This operation is used to terminate and remove an individual active audio call text message, and to return the final status.

6.3.6.1 Example: Terminating an active audio call text message (Informative)

6.3.6.1.1 Request

```
DELETE /exampleAPI/audiocall/v1/messages/text/msg123 HTTP/1.1
Accept: application/xml
Host: example.com
```

6.3.6.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:textMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
    <callSessionIdentifier>A45678</callSessionIdentifier>
    <callParticipant>tel:+19585550101</callParticipant>
    <callParticipant>tel:+19585550102</callParticipant>
    <text xml:lang="en">Welcome to the telephone conference</text>
    <messageStatusList>
      <messageStatus>
        <callParticipant>tel:+19585550101</callParticipant>
        <status>Played</status>
      </messageStatus>
      <messageStatus>
        <callParticipant>tel:+19585550102</callParticipant>
        <status>Terminated</status>
      </messageStatus>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList</resourceURL>
    </messageStatusList>
    <clientCorrelator>12345</clientCorrelator>
    <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123</resourceURL>
</ac:textMessage>
```

6.4 Resource: Individual text message status

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages/text/{messageId}/statusList

This resource is used to retrieve the status of an individual active audio call text message.

6.4.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.
messageId	identifier of the message resource.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.4.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.4.3 GET

This operation is used to retrieve the status of an individual active audio call text message.

6.4.3.1 Example: Retrieving status of an active audio call text message (Informative)

6.4.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages/text/msg123/statusList HTTP/1.1
Accept: application/xml
Host: example.com
```

6.4.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:messageStatusList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <messageStatus>
```



```
      <callParticipant>tel:+19585550101</callParticipant>
      <status>Played</status>
   </messageStatus>
   <messageStatus>
      <callParticipant>tel:+19585550102</callParticipant>
      <status>Pending</status>
   </messageStatus>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList</resourceURL>
</ac:messageStatusList>
```

6.4.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.4.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.4.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.5 Resource: Audio messages

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages/audio

This resource is used to create an audio call using audio content as the media, and to retrieve a list of all active audio call audio messages.

6.5.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.5.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.5.3 GET

This operation is used for reading a list of all active audio call audio messages.

6.5.3.1 Example: Retrieving a list of all active audio call text messages (Informative)

6.5.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages/audio HTTP/1.1
Accept: application/xml
Host: example.com
```

6.5.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:messageList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
  <audioMessage>
      <callSessionIdentifier>B45678</callSessionIdentifier>
     <callParticipant>tel:+19585550101</callParticipant>
     <callParticipant>tel:+19585550102</callParticipant>
      <mediaUrl>http://www.example.com/ann1.mp3</mediaUrl>
            <mediaType>audio/mpeg</mediaType>
       <messageStatusList>
         <messageStatus>
            <callParticipant>tel:+19585550101</callParticipant>
            <status>Played</status>
         </messageStatus>
         <messageStatus>
            <callParticipant>tel:+19585550102</callParticipant>
            <status>Pending</status>
         </messageStatus>
         <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList</resourceURL>
      </messageStatusList>
     <clientCorrelator>22345</clientCorrelator>
     <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123</resourceURL>
  </audioMessage>
   <audioMessage>
      <callSessionIdentifier>C45678</callSessionIdentifier>
      <callParticipant>tel:+19585550103</callParticipant>
      <callParticipant>tel:+19585550104</callParticipant>
      <mediaUrl>http://www.example.com/ann2.mp3</mediaUrl>
      <mediaType>audio/mpeg</mediaType>
      <messageStatusList>
         <messageStatus>
            <callParticipant>tel:+19585550103</callParticipant>
            <status>Played</status>
         </messageStatus>
         <messageStatus>
            <callParticipant>tel:+19585550104</callParticipant>
            <status>Playing</status>
         </messageStatus>
```

```
        <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg456/statusList</resourceURL>
      </messageStatusList>
      <clientCorrelator>32345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg456</resourceURL>
   </audioMessage>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio</resourceURL>
</ac:messageList>
```

6.5.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, POST' field in the response as per section 14.7 of [RFC2616].

6.5.5 POST

This operation is used to create a new audio message to be played to call participants.

6.5.5.1 Example 1: Creating an audio call audio message, using tel URI (Informative)

6.5.5.1.1 Request

```
POST /exampleAPI/audiocall/v1/messages/audio HTTP/1.1
Accept: application/xml
Content-Length: nnnn
Content-Type: application/xml
Host: example.com

<?xml version="1.0" encoding="UTF-8"?>
<ac:audioMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>B45678</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <callParticipant>tel:+19585550102</callParticipant>
   <mediaUrl>http://www.example.com/ann1.mp3</mediaUrl>
   <mediaType>audio/mpeg</mediaType>
   <clientCorrelator>22345</clientCorrelator>
 </ac:audioMessage>
```

Note that instead of the 'callSessionIdentifier' element, a 'link' element can be provided that points to the REST representation of the call session.

6.5.5.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:audioMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
<callSessionIdentifier>B45678</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <callParticipant>tel:+19585550102</callParticipant>
```

```
    <mediaUrl>http://www.example.com/ann1.mp3</mediaUrl>
    <mediaType>audio/mpeg</mediaType>
    <messageStatusList>
      <messageStatus>
        <callParticipant>tel:+19585550101</callParticipant>
        <status>Pending</status>
      </messageStatus>
      <messageStatus>
        <callParticipant>tel:+19585550102</callParticipant>
        <status>Pending</status>
      </messageStatus>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList</resourceURL>
    </messageStatusList>
    <clientCorrelator>22345</clientCorrelator>
    <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123</resourceURL>
</ac:audioMessage>
```

Note that alternatively, a ‘resourceReference’ root element can be returned, as illustrated in section 6.2.5.2.2.

6.5.5.2 Example 2: Creating an audio call audio message, using ACR (Informative)

6.5.5.2.1 Request

```
POST /exampleAPI/audiocall/v1/messages/audio HTTP/1.1
Accept: application/xml
Content-Length: nnnn
Content-Type: application/xml
Host: example.com

<?xml version="1.0" encoding="UTF-8"?>
<ac:audioMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
  <callSessionIdentifier>B45678</callSessionIdentifier>
  <callParticipant>acr:pseudonym123</callParticipant>
  <callParticipant>acr:pseudonym456</callParticipant>
  <mediaUrl>http://www.example.com/ann1.mp3</mediaUrl>
  <mediaType>audio/mpeg</mediaType>
  <clientCorrelator>22345</clientCorrelator>
</ac:audioMessage>
```

Note that instead of the ’callSessionIdentifier’ element, a ’link’ element can be provided that points to the REST representation of the call session.

6.5.5.2.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:audioMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
```



```
<callSessionIdentifier>B45678</callSessionIdentifier>
      <callParticipant>acr:pseudonym123</callParticipant>
      <callParticipant>acr:pseudonym456</callParticipant>
       <mediaUrl>http://www.example.com/ann1.mp3</mediaUrl>
       <mediaType>audio/mpeg</mediaType>
        <messageStatusList>
          <messageStatus>
             <callParticipant>acr:pseudonym123</callParticipant>
             <status>Pending</status>
          </messageStatus>
          <messageStatus>
             <callParticipant>acr:pseudonym456</callParticipant>
             <status>Pending</status>
          </messageStatus>
          <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList</resourceURL>
       </messageStatusList>
      <clientCorrelator>22345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123</resourceURL>
 </ac:audioMessage>
```

Note that alternatively, a 'resourceReference' root element can be returned, as illustrated in section 6.2.5.2.2.

6.5.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, POST' field in the response as per section 14.7 of [RFC2616].

6.6 Resource: Individual audio message

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages/audio/{messageId}

This resource is used to retrieve or terminate an individual active audio call audio message.

6.6.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.
messageId	identifier of the message resource.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.6.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.6.3 GET

This operation is used to retrieve an individual active audio call audio message.

6.6.3.1 Example: Retrieving an active audio call audio message (Informative)

6.6.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages/audio/msg123 HTTP/1.1
Accept: application/xml
Host: example.com
```

6.6.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:audioMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>B45678</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <callParticipant>tel:+19585550102</callParticipant>
   <mediaUrl>http://www.example.com/ann1.mp3</mediaUrl>
      <mediaType>audio/mpeg</mediaType>
   <messageStatusList>
      <messageStatus>
         <callParticipant>tel:+19585550101</callParticipant>
         <status>Played</status>
      </messageStatus>
      <messageStatus>
         <callParticipant>tel:+19585550102</callParticipant>
         <status>Pending</status>
      </messageStatus>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList</resourceURL>
   </messageStatusList>
   <clientCorrelator>22345</clientCorrelator>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123</resourceURL>
</ac:audioMessage>
```

6.6.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the ‘Allow: GET, DELETE’ field in the response as per section 14.7 of [RFC2616].

6.6.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the ‘Allow: GET, DELETE’ field in the response as per section 14.7 of [RFC2616].

6.6.6 DELETE

This operation is used to terminate and remove an individual active audio call audio message, and to return the final status..

6.6.6.1 Example: Terminating an active audio call audio message (Informative)

6.6.6.1.1 Request

```
DELETE /exampleAPI/audiocall/v1/messages/audio/msg123 HTTP/1.1
Accept: application/xml
Host: example.com
```

6.6.6.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:audioMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>B45678</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <callParticipant>tel:+19585550102</callParticipant>
   <mediaUrl>http://www.example.com/ann1.mp3</mediaUrl>
   <mediaType>audio/mpeg</mediaType>
   <messageStatusList>
      <messageStatus>
         <callParticipant>tel:+19585550101</callParticipant>
         <status>Played</status>
      </messageStatus>
      <messageStatus>
         <callParticipant>tel:+19585550102</callParticipant>
         <status>Terminated</status>
      </messageStatus>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList</resourceURL>
   </messageStatusList>
   <clientCorrelator>22345</clientCorrelator>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123</resourceURL>
</ac:audioMessage>
```

6.7 Resource: Individual audio message status

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages/audio/{messageId}/statusList

This resource is used to retrieve the status of individual active audio call audio message.

6.7.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description

serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.
messageId	identifier of the message resource.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.7.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.7.3 GET

This operation is used to retrieve the status of an individual active audio call audio message.

6.7.3.1 Example: Retrieving an active audio call audio message status (Informative)

6.7.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages/audio/msg123/statusList HTTP/1.1
Accept: application/xml
Host: example.com
```

6.7.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:messageStatusList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
      <messageStatus>
         <callParticipant>tel:+19585550101</callParticipant>
         <status>Played</status>
       </messageStatus>
       <messageStatus>
         <callParticipant>tel:+19585550102</callParticipant>
         <status>Pending</status>
       </messageStatus>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList</resourceURL>
</ac:messageStatusList>
```

6.7.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.7.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.7.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.8 Resource: VoiceXML messages

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages/voiceXml

This resource is used to create an audio call using VoiceXML content as the media, and to retrieve a list of all active audio call VoiceXML messages.

6.8.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.8.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.8.3 GET

This operation is used for reading a list of all active audio call VoiceXML messages.

6.8.3.1 Example: Retrieving a list of all active audio call VoiceXML messages (Informative)

6.8.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages/voiceXml HTTP/1.1
Accept: application/xml
Host: example.com
```


6.8.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:messageList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
  <voiceXmlMessage>
      <callSessionIdentifier>B45678</callSessionIdentifier>
     <callParticipant>tel:+19585550101</callParticipant>
     <callParticipant>tel:+19585550102</callParticipant>
      <mediaUrl>http://www.example.com/ann1.vxml</mediaUrl>
       <messageStatusList>
          <messageStatus>
             <callParticipant>tel:+19585550101</callParticipant>
             <status>Played</status>
          </messageStatus>
          <messageStatus>
             <callParticipant>tel:+19585550102</callParticipant>
             <status>Pending</status>
          </messageStatus>
          <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList</resourceURL>
       </messageStatusList>
     <clientCorrelator>42345</clientCorrelator>
     <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123</resourceURL>
  </voiceXmlMessage >
  <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml</resourceURL>
</ac:messageList>
```

6.8.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, POST' field in the response as per section 14.7 of [RFC2616].

6.8.5 POST

This operation is used to create a new VoiceXML message to be played to call participants.

6.8.5.1 Example: Creating an audio call VoiceXML message (Informative)

6.8.5.1.1 Request

```
POST /exampleAPI/audiocall/v1/messages/voiceXml HTTP/1.1
Accept: application/xml
Content-Type: application/xml
Content-Length: nnnn
Host: example.com

<?xml version="1.0" encoding="UTF-8"?>
<ac:voiceXMLMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>D45678</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <callParticipant>tel:+19585550102</callParticipant>
```



```
   <mediaUrl>http://www.example.com/ann1.vxml</mediaUrl>
   <clientCorrelator>42345</clientCorrelator>
 </ac:voiceXMLMessage>
```

6.8.5.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:voiceXMLMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
      <callSessionIdentifier>D45678</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <callParticipant>tel:+19585550102</callParticipant>
       <mediaUrl>http://www.example.com/ann1.vxml</mediaUrl>
        <messageStatusList>
          <messageStatus>
             <callParticipant>tel:+19585550101</callParticipant>
             <status>Pending</status>
          </messageStatus>
          <messageStatus>
             <callParticipant>tel:+19585550102</callParticipant>
             <status>Pending</status>
          </messageStatus>
          <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList</resourceURL>
       </messageStatusList>
      <clientCorrelator>42345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123</resourceURL>
 </ac:voiceXMLMessage>
```

Note that alternatively, a ‘resourceReference’ root element can be returned, as illustrated in section 6.2.5.2.2.

6.8.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the ‘Allow: GET, POST’ field in the response as per section 14.7 of [RFC2616].

6.9 Resource: Individual VoiceXML message

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages/voiceXml/{messageId}

This resource is used to retrieve or terminate an individual active audio call VoiceXML message.

6.9.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.
messageId	identifier of the message resource.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.9.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.9.3 GET

This operation is used to retrieve an individual active audio call VoiceXML message.

6.9.3.1 Example: Retrieving an active audio call VoiceXML message(Informative)

6.9.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages/voiceXml/msg123 HTTP/1.1
Accept: application/xml
Host: example.com
```

6.9.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:voiceXMLMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>D45678</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <callParticipant>tel:+19585550102</callParticipant>
   <mediaUrl>http://www.example.com/ann1.vxml</mediaUrl>
   <messageStatusList>
      <messageStatus>
         <callParticipant>tel:+19585550101</callParticipant>
         <status>Played</status>
      </messageStatus>
      <messageStatus>
         <callParticipant>tel:+19585550102</callParticipant>
         <status>Pending</status>
      </messageStatus>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList</resourceURL>
   </messageStatusList>
   <clientCorrelator>42345</clientCorrelator>
```



```
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123</resourceURL>
</ac:voiceXMLMessage>
```

6.9.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, DELETE' field in the response as per section 14.7 of [RFC2616].

6.9.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, DELETE' field in the response as per section 14.7 of [RFC2616].

6.9.6 DELETE

This operation is used to terminate and remove an individual active audio call VoiceXML message, and to return the final status.

6.9.6.1 Example: Terminating an active audio call VoiceXML message (Informative)

6.9.6.1.1 Request

```
DELETE /exampleAPI/audiocall/v1/messages/voiceXml/msg123 HTTP/1.1
Accept: application/xml
Host: example.com
```

6.9.6.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:voiceXMLMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>D45678</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <callParticipant>tel:+19585550102</callParticipant>
   <mediaUrl>http://www.example.com/ann1.vxml</mediaUrl>
   <messageStatusList>
      <messageStatus>
         <callParticipant>tel:+19585550101</callParticipant>
         <status>Played</status>
      </messageStatus>
      <messageStatus>
         <callParticipant>tel:+19585550102</callParticipant>
         <status>Terminated</status>
      </messageStatus>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList</resourceURL>
   </messageStatusList>
   <clientCorrelator>42345</clientCorrelator>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123</resourceURL>
</ac:voiceXMLMessage>
```


6.10 Resource: Individual voiceXML message status

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages/voiceXml/{messageId}/statusList

This resource is used to retrieve the status of individual active audio call VoiceXML message.

6.10.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.
messageId	identifier of the message resource.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.10.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.10.3 GET

This operation is used to retrieve the status of an individual active audio call VoiceXML message.

6.10.3.1 Example: Retrieving status of an active audio call VoiceXML message (Informative)

6.10.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList HTTP/1.1
Accept: application/xml
Host: example.com
```

6.10.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:messageStatusList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
    <messageStatus>
```



```
        <callParticipant>tel:+19585550101</callParticipant>
        <status>Played</status>
      </messageStatus>
      <messageStatus>
        <callParticipant>tel:+19585550102</callParticipant>
        <status>Pending</status>
      </messageStatus>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList</resourceURL>
</ac:messageStatusList>
```

6.10.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.10.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.10.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.11 Resource: Video messages

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages/video

This resource is used to create an audio call using video content as the media, and to retrieve a list of all active audio call video messages.

6.11.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.11.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.11.3 GET

This operation is used for reading a list of all active audio call video messages.

6.11.3.1 Example: Retrieving a list of all active audio call video messages (Informative)

6.11.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages/video HTTP/1.1
Accept: application/xml
Host: example.com
```

6.11.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:messageList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
  <videoMessage>
      <callSessionIdentifier>E45678</callSessionIdentifier>
     <callParticipant>tel:+19585550101</callParticipant>
     <callParticipant>tel:+19585550102</callParticipant>
      <mediaUrl>http://www.example.com/ann1.mp4</mediaUrl>
      <mediaType>video/mp4</mediaType>
       <messageStatusList>
         <messageStatus>
            <callParticipant>tel:+19585550101</callParticipant>
            <status>Played</status>
         </messageStatus>
         <messageStatus>
            <callParticipant>tel:+19585550102</callParticipant>
            <status>Pending</status>
         </messageStatus>
         <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList</resourceURL>
      </messageStatusList>
     <clientCorrelator>42345</clientCorrelator>
     <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123</resourceURL>
  </videoMessage >
  <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video</resourceURL>
</ac:messageList>
```

6.11.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, POST' field in the response as per section 14.7 of [RFC2616].

6.11.5 POST

This operation is used to create a new video message to be played to call participants.

6.11.5.1 Example: Creating an audio call video message (Informative)

6.11.5.1.1 Request

```
POST /exampleAPI/audiocall/v1/messages/video HTTP/1.1
Accept: application/xml
Content-Type: application/xml
Content-Length: nnnn
Host: example.com

<?xml version="1.0" encoding="UTF-8"?>
<ac:videoMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>E45678</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <callParticipant>tel:+19585550102</callParticipant>
   <mediaUrl>http://www.example.com/ann1.mp4</mediaUrl>
   <mediaType>video/mp4</mediaType>
   <clientCorrelator>52345</clientCorrelator>
 </ac:videoMessage>
```

6.11.5.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/messages/video/msg123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:videoMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
      <callSessionIdentifier>E45678</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <callParticipant>tel:+19585550102</callParticipant>
       <mediaUrl>http://www.example.com/ann1.mp4</mediaUrl>
       <mediaType>video/mp4</mediaType>
        <messageStatusList>
          <messageStatus>
             <callParticipant>tel:+19585550101</callParticipant>
             <status>Pending</status>
          </messageStatus>
          <messageStatus>
             <callParticipant>tel:+19585550102</callParticipant>
             <status>Pending</status>
          </messageStatus>
          <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList</resourceURL>
       </messageStatusList>
      <clientCorrelator>52345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123</resourceURL>
 </ac:videoMessage>
```

Note that alternatively, a ‘resourceReference’ root element can be returned, as illustrated in section 6.2.5.2.2.

6.11.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, POST' field in the response as per section 14.7 of [RFC2616].

6.12 Resource: Individual video message

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages/video/{messageId}

This resource is used to retrieve or terminate an individual active audio call video message.

6.12.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.
messageId	identifier of the message resource.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.12.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.12.3 GET

This operation is used to retrieve an individual active audio call video message.

6.12.3.1 Example: Retrieving an active audio call video message (Informative)

6.12.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages/video/msg123 HTTP/1.1
Accept: application/xml
Host: example.com
```

6.12.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
```

```
<ac:videoMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>E45678</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <callParticipant>tel:+19585550102</callParticipant>
   <mediaUrl>http://www.example.com/ann1.mp4</mediaUrl>
   <mediaType>video/mp4</mediaType>
   <messageStatusList>
      <messageStatus>
         <callParticipant>tel:+19585550101</callParticipant>
         <status>Played</status>
      </messageStatus>
      <messageStatus>
         <callParticipant>tel:+19585550102</callParticipant>
         <status>Pending</status>
      </messageStatus>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList</resourceURL>
   </messageStatusList>
   <clientCorrelator>52345</clientCorrelator>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123</resourceURL>
</ac:videoMessage>
```

6.12.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, DELETE' field in the response as per section 14.7 of [RFC2616].

6.12.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, DELETE' field in the response as per section 14.7 of [RFC2616].

6.12.6 DELETE

This operation is used to terminate and remove an individual active audio call video message, and to return the final status.

6.12.6.1 Example: Terminating an active audio call video message (Informative)

6.12.6.1.1 Request

```
DELETE /exampleAPI/audiocall/v1/messages/video/msg123 HTTP/1.1
Accept: application/xml
Host: example.com
```

6.12.6.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:videoMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>E45678</callSessionIdentifier>
```

```
   <callParticipant>tel:+19585550101</callParticipant>
   <callParticipant>tel:+19585550102</callParticipant>
   <mediaUrl>http://www.example.com/ann1.mp4</mediaUrl>
   <mediaType>video/mp4</mediaType>
   <messageStatusList>
      <messageStatus>
         <callParticipant>tel:+19585550101</callParticipant>
         <status>Played</status>
      </messageStatus>
      <messageStatus>
         <callParticipant>tel:+19585550102</callParticipant>
         <status>Terminated</status>
      </messageStatus>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList</resourceURL>
   </messageStatusList>
   <clientCorrelator>52345</clientCorrelator>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123</resourceURL>
</ac:videoMessage>
```

6.13 Resource: Individual video message status

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/messages/video/{messageId}/statusList

This resource is used to retrieve the status of individual active audio call video message.

6.13.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.
messageId	identifier of the message resource.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.13.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.13.3 GET

This operation is used to retrieve the status of an individual active audio call video message.

6.13.3.1 Example: Retrieving status of an active audio call video message (Informative)

6.13.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/messages/video/msg123/statusList HTTP/1.1
Accept: application/xml
Host: example.com
```

6.13.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:messageStatusList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
     <messageStatus>
        <callParticipant>tel:+19585550101</callParticipant>
        <status>Played</status>
      </messageStatus>
      <messageStatus>
        <callParticipant>tel:+19585550102</callParticipant>
        <status>Pending</status>
      </messageStatus>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList</resourceURL>
</ac:messageStatusList>
```

6.13.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the ‘Allow: GET’ field in the response as per section 14.7 of [RFC2616].

6.13.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the ‘Allow: GET’ field in the response as per section 14.7 of [RFC2616].

6.13.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the ‘Allow: GET’ field in the response as per section 14.7 of [RFC2616].

6.14 Resource: All media capture interactions

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/interactions

This resource is used to obtain all active media capture interactions.

6.14.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.14.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.14.3 GET

This operation is used for reading a list of all active media capture interactions.

6.14.3.1 Example: Retrieving a list of all active media capture interactions (Informative)

6.14.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/interactions HTTP/1.1
Accept: application/xml
Host: example.com
```

6.14.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:interactionList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <digitCapture>
      <callSessionIdentifier>F14567</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <playingConfiguration>
         <playFileLocation>http://www.example.com/msg1.mp3</playFileLocation>
         <messageFormat>Audio</messageFormat>
        <mediaType>audio/mpeg</mediaType>
         <interruptMedia>false</interruptMedia>
      </playingConfiguration>
      <digitConfiguration>
```



```
        <minDigits>1</minDigits>
        <maxDigits>1</maxDigits>
        <endChar>#</endChar>
      </digitConfiguration>
      <clientCorrelator>62345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123</resourceURL>
    </digitCapture>
    <digitCapture>
      <callSessionIdentifier>F24567</callSessionIdentifier>
      <callParticipant>tel:+19585550104</callParticipant>
      <playingConfiguration>
        <playFileLocation>http://www.example.com/msg3.mp3</playFileLocation>
        <messageFormat>Audio</messageFormat>
        <mediaType>audio/mpeg</mediaType>
        <interruptMedia>false</interruptMedia>
      </playingConfiguration>
      <digitConfiguration>
        <minDigits>1</minDigits>
        <maxDigits>1</maxDigits>
        <endChar>#</endChar>
      </digitConfiguration>
      <clientCorrelator>72345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/collection/int456</resourceURL>
    </digitCapture>
    <recordingCapture>
      <callSessionIdentifier>F34567</callSessionIdentifier>
      <callParticipant>tel:+19585550102</callParticipant>
      <playingConfiguration>
        <playFileLocation>http://www.example.com/msg2.mp3</playFileLocation>
        <messageFormat>Audio</messageFormat>
        <mediaType>audio/mpeg</mediaType>
        <interruptMedia>false</interruptMedia>
      </playingConfiguration>
      <recordingConfiguration>
        <recFileLocation>http://www.example.com/rec1.mp3</recFileLocation>
        <maxRecordingLength>10</maxRecordingLength>
        <endChar>#</endChar>
      </recordingConfiguration>
      <clientCorrelator>82345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/recording/int123</resourceURL>
    </recordingCapture>
    <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions</resourceURL>
</ac:interactionList>
```

6.14.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the ‘Allow: GET’ field in the response as per section 14.7 of [RFC2616].

6.14.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the ‘Allow: GET’ field in the response as per section 14.7 of [RFC2616].

6.14.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET' field in the response as per section 14.7 of [RFC2616].

6.15 Resource: Play-and-collect interactions

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/interactions/collection

This resource is used to play a media file to call participants and collect digits. It is also used to retrieve a list of all calls where media is being played to participants and digits collected.

6.15.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.15.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.15.3 GET

This operation is used for reading the list of calls where media is being played and digits collected.

6.15.3.1 Example: Retrieving a list of all play-and-collect interactions (Informative)

6.15.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/interactions/collection HTTP/1.1
Accept: application/xml
Host: example.com
```

6.15.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:interactionList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
```

```
  <digitCapture>
    <callSessionIdentifier>F14567</callSessionIdentifier>
    <callParticipant>tel:+19585550101</callParticipant>
    <playingConfiguration>
      <playFileLocation>http://www.example.com/msg1.mp3</playFileLocation>
      <messageFormat>Audio</messageFormat>
      <mediaType>audio/mpeg</mediaType>
      <interruptMedia>false</interruptMedia>
    </playingConfiguration>
    <digitConfiguration>
      <minDigits>1</minDigits>
      <maxDigits>1</maxDigits>
      <endChar>#</endChar>
    </digitConfiguration>
    <clientCorrelator>62345</clientCorrelator>
    <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123</resourceURL>
  </digitCapture>
  <digitCapture>
    <callSessionIdentifier>F24567</callSessionIdentifier>
    <callParticipant>tel:+19585550104</callParticipant>
    <playingConfiguration>
      <playFileLocation>http://www.example.com/msg3.mp3</playFileLocation>
      <messageFormat>Audio</messageFormat>
      <mediaType>audio/mpeg</mediaType>
      <interruptMedia>false</interruptMedia>
    </playingConfiguration>
    <digitConfiguration>
      <minDigits>1</minDigits>
      <maxDigits>1</maxDigits>
      <endChar>#</endChar>
    </digitConfiguration>
    <clientCorrelator>72345</clientCorrelator>
    <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/collection/int456</resourceURL>
  </digitCapture>
  <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/collection</resourceURL>
</ac:interactionList>
```

6.15.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, POST' field in the response as per section 14.7 of [RFC2616].

6.15.5 POST

This operation is used to play a media file to call participants and collect digits from the participants.

6.15.5.1 Example 1: Playing a media file and collecting digits, returning a copy of the created resource (Informative)

6.15.5.1.1 Request

```
POST /exampleAPI/audiocall/v1/interactions/collection HTTP/1.1
Accept: application/xml
Content-Type: application/xml
```



```
Content-Length: nnnn
Host: example.com

<?xml version="1.0" encoding="UTF-8"?>
<ac:digitCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>F14567</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <playingConfiguration>
      <playFileLocation>http://www.example.com/msg1.mp3</playFileLocation>
      <messageFormat>Audio</messageFormat>
      <mediaType>audio/mpeg</mediaType>
      <interruptMedia>false</interruptMedia>
   </playingConfiguration>
   <digitConfiguration>
      <minDigits>1</minDigits>
      <maxDigits>1</maxDigits>
      <endChar>#</endChar>
   </digitConfiguration>
   <clientCorrelator>62345</clientCorrelator>
</ac:digitCapture>
```

Note that instead of the 'callSessionIdentifier' element, a 'link' element can be provided that points to the REST representation of the call session.

6.15.5.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:digitCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>F14567</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <playingConfiguration>
      <playFileLocation>http://www.example.com/msg1.mp3</playFileLocation>
      <messageFormat>Audio</messageFormat>
      <mediaType>audio/mpeg</mediaType>
      <interruptMedia>false</interruptMedia>
   </playingConfiguration>
   <digitConfiguration>
      <minDigits>1</minDigits>
      <maxDigits>1</maxDigits>
      <endChar>#</endChar>
   </digitConfiguration>
   <clientCorrelator>62345</clientCorrelator>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123</resourceURL>
</ac:digitCapture>
```

6.15.5.2 Example 2: Playing a media file and collecting digits, returning the location of the created resource (Informative)

6.15.5.2.1 Request

```
POST /exampleAPI/audiocall/v1/ interactions/collection HTTP/1.1
Accept: application/xml
Content-Type: application/xml
Content-Length: nnnn
Host: example.com

<?xml version="1.0" encoding="UTF-8"?>
<ac:digitCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>F14567</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <playingConfiguration>
      <playFileLocation>http://www.example.com/msg1.mp3</playFileLocation>
      <messageFormat>Audio</messageFormat>
      <mediaType>audio/mpeg</mediaType>
      <interruptMedia>false</interruptMedia>
   </playingConfiguration>
   <digitConfiguration>
      <minDigits>1</minDigits>
      <maxDigits>1</maxDigits>
      <endChar>#</endChar>
   </digitConfiguration>
   <clientCorrelator>62345</clientCorrelator>
</ac:digitCapture>
```

6.15.5.2.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<common:resourceReference xmlns:common="urn:oma:xml:rest:netapi:common:1">
  <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123</resourceURL>
</common:resourceReference>
```

6.15.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, POST' field in the response as per section 14.7 of [RFC2616].

6.16 Resource: Individual play-and-collect interaction

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/interactions/collection/{interactionId}

This resource is used to retrieve or terminate an individual play-and-collect interaction.

6.16.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.
interactionId	identifier of the interaction resource.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.16.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.16.3 GET

This operation is used to retrieve an individual play-and-collect interaction.

6.16.3.1 Example: Retrieving an individual play-and-collect interaction (Informative)

6.16.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/interactions/collection/int123 HTTP/1.1
Accept: application/xml
Host: example.com
```

6.16.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:digitCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>F14567</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <playingConfiguration>
      <playFileLocation>http://www.example.com/msg1.mp3</playFileLocation>
      <messageFormat>Audio</messageFormat>
      <mediaType>audio/mpeg</mediaType>
      <interruptMedia>false</interruptMedia>
   </playingConfiguration>
   <digitConfiguration>
      <minDigits>1</minDigits>
      <maxDigits>1</maxDigits>
      <endChar>#</endChar>
```



```
   </digitConfiguration>
   <clientCorrelator>62345</clientCorrelator>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123</resourceURL>
</ac:digitCapture>
```

6.16.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, DELETE' field in the response as per section 14.7 of [RFC2616].

6.16.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, DELETE' field in the response as per section 14.7 of [RFC2616].

6.16.6 DELETE

This operation is used to terminate and remove an individual play-and-collect interaction.

6.16.6.1 Example: Stopping interaction and removing information (Informative)

6.16.6.1.1 Request

```
DELETE /exampleAPI/audiocall/v1/interactions/collection/int123 HTTP/1.1
Accept: application/xml
Host: example.com
```

6.16.6.1.2 Response

```
HTTP/1.1 204 No Content
Date: Mon, 28 Jun 2010 17:51:59 GMT
```

6.17 Resource: Play media and record participant response

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/interactions/recording

This resource is used to play a media file to call participants and record a response. It is also used to retrieve a list of all calls where media is being played to participants and responses recorded.

6.17.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.17.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.17.3 GET

This operation is used for reading the list of calls where media is being played and responses collected.

6.17.3.1 Example: Retrieving a list of all play-and-record interactions (Informative)

6.17.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/interactions/recording HTTP/1.1
Accept: application/xml
Host: example.com
```

6.17.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:interactionList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
  <recordingCapture>
     <callSessionIdentifier>F34567</callSessionIdentifier>
     <callParticipant>tel:+19585550102</callParticipant>
     <playingConfiguration>
        <playFileLocation>http://www.example.com/msg2.mp3</playFileLocation>
        <messageFormat>Audio</messageFormat>
        <mediaType>audio/mpeg</mediaType>
        <interruptMedia>false</interruptMedia>
     </playingConfiguration>
     <recordingConfiguration>
        <recFileLocation>http://www.example.com/rec2.mp3</recFileLocation>
        <maxRecordingLength>10</maxRecordingLength>
        <endChar>#</endChar>
     </recordingConfiguration>
     <clientCorrelator>82345</clientCorrelator>
     <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/recording/int123</resourceURL>
  </recordingCapture>
  <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions</resourceURL>
</ac:interactionList>
```

6.17.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the ‘Allow: GET, POST’ field in the response as per section 14.7 of [RFC2616].

6.17.5 POST

This operation is used to play a media file to call participants and record a response from the participants.

6.17.5.1 Example: Playing a media file and recording response (Informative)

6.17.5.1.1 Request

```
POST /exampleAPI/audiocall/v1/interactions/recording HTTP/1.1
Accept: application/xml
Content-Type: application/xml
Content-Length: nnnn
Host: example.com

<?xml version="1.0" encoding="UTF-8"?>
<ac:recordingCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>F34567</callSessionIdentifier>
   <callParticipant>tel:+19585550102</callParticipant>
   <playingConfiguration>
      <playFileLocation>http://www.example.com/msg2.mp3</playFileLocation>
      <messageFormat>Audio</messageFormat>
      <mediaType>audio/mpeg</mediaType>
      <interruptMedia>false</interruptMedia>
   </playingConfiguration>
   <recordingConfiguration>
      <recFileLocation>http://www.example.com/rec2.mp3</recFileLocation>
      <maxRecordingLength>10</maxRecordingLength>
      <endChar>#</endChar>
   </recordingConfiguration>
   <clientCorrelator>82345</clientCorrelator>
</ac:recordingCapture>
```

6.17.5.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/interactions/recording/int123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:recordingCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>F34567</callSessionIdentifier>
   <callParticipant>tel:+19585550102</callParticipant>
   <playingConfiguration>
      <playFileLocation>http://www.example.com/msg2.mp3</playFileLocation>
      <messageFormat>Audio</messageFormat>
      <mediaType>audio/mpeg</mediaType>
      <interruptMedia>false</interruptMedia>
   </playingConfiguration>
   <recordingConfiguration>
      <recFileLocation>http://www.example.com/rec2.mp3</recFileLocation>
      <maxRecordingLength>10</maxRecordingLength>
      <endChar>#</endChar>
   </recordingConfiguration>
   <clientCorrelator>82345</clientCorrelator>
```

```
    <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/recording/int123</resourceURL>
</ac:recordingCapture>
```

Note that alternatively, a 'resourceReference' root element can be returned, as illustrated in section 6.15.5.2.2.

6.17.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, POST' field in the response as per section 14.7 of [RFC2616].

6.18 Resource: Individual play-and-record interaction

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/interactions/recording/{interactionId}

This resource is used to retrieve or terminate an individual play-and-record interaction.

6.18.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.
interactionId	identifier of the interaction resource.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.18.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.18.3 GET

This operation is used to retrieve an individual play and record interaction.

6.18.3.1 Example: Retrieving an individual play-and-record interaction (Informative)

6.18.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/interactions/recording/int123 HTTP/1.1
Accept: application/xml
Host: example.com
```


6.18.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:recordingCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>F34567</callSessionIdentifier>
   <callParticipant>tel:+19585550102</callParticipant>
   <playingConfiguration>
      <playFileLocation>http://www.example.com/msg2.mp3</playFileLocation>
      <messageFormat>Audio</messageFormat>
      <mediaType>audio/mpeg</mediaType>
      <interruptMedia>false</interruptMedia>
   </playingConfiguration>
   <recordingConfiguration>
      <recFileLocation>http://www.example.com/rec2.mp3</recFileLocation>
     <maxRecordingLength>10</maxRecordingLength>
     <endChar>#</endChar>
   </recordingConfiguration>
   <clientCorrelator>82345</clientCorrelator>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/recording/int123</resourceURL>
</ac:recordingCapture>
```

6.18.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, DELETE' field in the response as per section 14.7 of [RFC2616].

6.18.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, DELETE' field in the response as per section 14.7 of [RFC2616].

6.18.6 DELETE

This operation is used to terminate and remove an individual play-and-record interaction.

6.18.6.1 Example: Stopping play-and-record interaction and removing information (Informative)

6.18.6.1.1 Request

```
DELETE /exampleAPI/audiocall/v1/interactions/recording/int123 HTTP/1.1
Accept: application/xml
Host: example.com
```

6.18.6.1.2 Response

```
HTTP/1.1 204 No Content
Date: Mon, 28 Jun 2010 17:51:59 GMT
```


6.19 Resource: Engage the participants in a dialog and automatically recognize their spoken responses

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/interactions/recognition

This resource is used to engage the participants in a dialog and automatically recognize their spoken responses. It is also used to retrieve a list of all calls where such a dialog is active.

6.19.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.19.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.19.3 GET

This operation is used for reading the list of calls where an automatic speech recognition dialog is active.

6.19.3.1 Example: Retrieving a list of all recognition interactions (Informative)

6.19.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/interactions/recognition HTTP/1.1
Accept: application/xml
Host: example.com
```

6.19.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:interactionList xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
  <recognitionCapture>
```



```
      <callSessionIdentifier>G34567</callSessionIdentifier>
      <callParticipant>tel:+19585550102</callParticipant>
      <asrConfiguration>
         <scriptFileLocation>http://www.example.com/asr.vxml</scriptFileLocation>
         <scriptType>application/voicexml+xml</scriptType>
         <endChar>#</endChar>
      </asrConfiguration>
      <clientCorrelator>92345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int123</resourceURL>
   </recognitionCapture>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions</resourceURL>
</ac:interactionList>
```

6.19.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, POST' field in the response as per section 14.7 of [RFC2616].

6.19.5 POST

This operation is used to engage the participants in a dialog and automatically recognize their spoken response.

6.19.5.1 Example: Engaging the participants in a dialog and automatically recognizing their spoken response; script addressed by URL (Informative)

In this example, the script that controls the dialog and recognition is passed via a URL.

6.19.5.1.1 Request

```
POST /exampleAPI/audiocall/v1/interactions/recognition HTTP/1.1
Accept: application/xml
Content-Type: application/xml
Content-Length: nnnn
Host: example.com

<?xml version="1.0" encoding="UTF-8"?>
<ac:recognitionCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
    <callSessionIdentifier>G34567</callSessionIdentifier>
    <callParticipant>tel:+19585550102</callParticipant>
    <asrConfiguration>
       <scriptFileLocation>http://www.example.com/asr.vxml</scriptFileLocation>
       <scriptType>application/voicexml+xml</scriptType>
       <endChar>#</endChar>
    </asrConfiguration>
    <clientCorrelator>92345</clientCorrelator>
</ac:recognitionCapture>
```

6.19.5.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int345
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT
```



```
<?xml version="1.0" encoding="UTF-8"?>
<ac:recognitionCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
    <callSessionIdentifier>G34567</callSessionIdentifier>
    <callParticipant>tel:+19585550102</callParticipant>
    <asrConfiguration>
       <scriptFileLocation>http://www.example.com/asr.vxml</scriptFileLocation>
       <scriptType>application/voicexml+xml</scriptType>
       <endChar>#</endChar>
    </asrConfiguration>
    <clientCorrelator>92345</clientCorrelator>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int123</resourceURL>
</ac:recognitionCapture>
```

Note that alternatively, a 'resourceReference' root element can be returned, as illustrated in section 6.19.5.2.

6.19.5.2 Example: Engaging the participants in a dialog and automatically recognizing their spoken response; script embedded (Informative)

In this example, the script that controls the dialog and recognition is embedded in the request body. Furthermore, a reference to the created resource is returned rather than a copy.

6.19.5.2.1 Request

```
POST /exampleAPI/audiocall/v1/interactions/recognition HTTP/1.1
Accept: application/xml
Content-Type: application/xml
Content-Length: nnnn
Host: example.com

<?xml version="1.0" encoding="UTF-8"?>
<ac:recognitionCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
    <callSessionIdentifier>H34567</callSessionIdentifier>
    <callParticipant>tel:+19585550102</callParticipant>
    <asrConfiguration>
       <script>
          <vxml xmlns="http://www.w3.org/2001/vxml">
             <!-- voiceXML instructions -->
          </vxml>
       </script>
       <scriptType>application/voicexml+xml</scriptType>
       <endChar>#</endChar>
    </asrConfiguration>
    <clientCorrelator>a2345</clientCorrelator>
</ac:recognitionCapture>
```

6.19.5.2.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int345
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<common:resourceReference xmlns:common="urn:oma:xml:rest:netapi:common:1">
```



```
  <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int345</resourceURL>
</common:resourceReference>
```

Note that alternatively, a copy of the created resource can be returned, as illustrated in section 6.19.3.1.

6.19.6 DELETE

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, POST' field in the response as per section 14.7 of [RFC2616].

6.20 Resource: Individual recognition interaction

The resource used is:

http://{serverRoot}/audiocall/{apiVersion}/interactions/recognition/{interactionId}

This resource is used to retrieve or terminate an speech recognition interaction.

6.20.1 Request URL variables

The following request URL variables are common for all HTTP commands:

Name	Description
serverRoot	server base url: hostname+port+base path. Example: example.com/exampleAPI
apiVersion	version of the API clients want to use. The value of this variable is defined in section 5.1.
interactionId	identifier of the interaction resource.

See section 6 for a statement on the escaping of reserved characters in URL variables.

6.20.2 Response Codes and Error Handling

For HTTP response codes, see [REST_NetAPI_Common].

For Policy Exception and Service Exception fault codes applicable to Audio Call, see section 0.

6.20.3 GET

This operation is used to retrieve an speech recognition interaction.

6.20.3.1 Example: Retrieving an individual speech recognition interaction (Informative)

6.20.3.1.1 Request

```
GET /exampleAPI/audiocall/v1/interactions/recognition/int123 HTTP/1.1
Accept: application/xml
Host: example.com
```


6.20.3.1.2 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:recognitionCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
    <callSessionIdentifier>G34567</callSessionIdentifier>
    <callParticipant>tel:+19585550102</callParticipant>
    <asrConfiguration>
       <scriptFileLocation>http://www.example.com/asr.vxml</scriptFileLocation>
       <scriptType>application/voicexml+xml</scriptType>
       <endChar>#</endChar>
    </asrConfiguration>
    <clientCorrelator>92345</clientCorrelator>
    <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int123</resourceURL>
</ac:recognitionCapture>
```

6.20.4 PUT

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, DELETE' field in the response as per section 14.7 of [RFC2616].

6.20.5 POST

Method not allowed by the resource. The returned HTTP error status is 405. The server SHOULD also include the 'Allow: GET, DELETE' field in the response as per section 14.7 of [RFC2616].

6.20.6 DELETE

This operation is used to terminate and remove an individual speech recognition interaction.

6.20.6.1 Example: Stopping speech recognition interaction and removing information (Informative)

6.20.6.1.1 Request

```
DELETE /exampleAPI/audiocall/v1/interactions/recognition/int123 HTTP/1.1
Accept: application/xml
Host: example.com
```

6.20.6.1.2 Response

```
HTTP/1.1 204 No Content
Date: Mon, 28 Jun 2010 17:51:59 GMT
```


7. Fault definitions

7.1 Service Exceptions

For common Service Exceptions refer to [REST_NetAPI_Common].

The following additional Service Exception codes are defined for the Audio Call API.

7.1.1 SVC0290: Duplicate media stream

Name	Description
MessageID	SVC0290
Text	Duplicate media stream
Variables	None
HTTP status code(s)	400 Bad request

7.1.2 SVC0291: Media stream does not match

Name	Description
MessageID	SVC0291
Text	Media stream does not match type specified
Variables	None
HTTP status code(s)	400 Bad request

7.2 Policy Exceptions

For common Policy Exceptions refer to [REST_NetAPI_Common].

There are no additional Policy Exception codes defined for the Audio Call API.

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
n/a	n/a	No prior version

A.2 Draft/Candidate Version 1.0 History

Document Identifier	Date	Sections	Description
Draft Versions OMA-TS-REST_NetAPI_AudioCall-V1_0	03 May 2011	Many	Structural changes to fit the OMA RESTful Network API release. This version inherits the technical content of OMA-TS-ParlayREST_AudioCall-V1_0-20110111-C and applies changes according to ARC INP 30R01, 98R02, 155R01,156R01, 69R01, 175R01, 186, 187R02 and 162R02
	23 May 2011	Many	Adapted to new REST NetAPI TS template OMA-TEMPLATE-Technical_Spec_RESTful_Network_API-20110502-D. OMA-ARC-REST-NetAPI-2011-0017R02-INP_ApiVersion_in_NetAPI_TSs implemented.
	15 Jul 2011	Many	OMA-ARC-REST-NetAPI-2011-0144-CR_AC_fictional_phone_numbers implemented
	27 Jul 2011	Many	OMA-ARC-REST-NetAPI-2011-0172-CR_AudioCall_tel_URI_fixes_and_Notif_channel_changes implemented
	08 Sep 2011	Annex B	OMA-ARC-REST-NetAPI-2011-0179R01-CR_Audio_Call_SCR_references implemented
	20 Sep 2011	Many	OMA-ARC-REST-NetAPI-2011-0248R01-CR_ACR_AudioCall implemented
	08 Nov 2011	Many	OMA-ARC-REST-NetAPI-2011-0301R02-CR_AudioCall_editorial_CONRR_resolution implemented.
	14 Nov 2011	Many	OMA-ARC-REST-NetAPI-2011-0371-CR_Audiocall_more_editorials implemented
	18 Nov 2011	Many	OMA-ARC-REST-NetAPI-2011-0377-CR_AudioCall_actions_and_editorials implemented
	30 Nov 2011	Annex G	Implemented - OMA-ARC-REST-NetAPI-2011-0432-CR_AudioCall_Annex_G - OMA-ARC-REST-NetAPI-2011-0424-INP_HTML_401_reference_blueprint
	16 Jan 2011	Many	OMA-ARC-REST-NetAPI-2012-0011R01-CR_TS_AudioCall_CONR_resolution implemented
	26 Jan 2012	Many	OMA-ARC-REST-NetAPI-2012-0024-CR_AudioCall_interruptMedia_fix implemented Editorial: Annex G converted to Landscape mode
	31 Jan 2012	Many	Implemented OMA-ARC-REST-NetAPI-2012-0033-CR_AudioCall_various_changes
	27 Feb 2012	Many	Implemented - OMA-ARC-REST-NetAPI-2012-0075R01-CR_AudioCall_CONR_A0029 - OMA-ARC-REST-NetAPI-2012-0078R01-CR_AudioCall_more_CONR_resolutions
	07 Mar 2012	Many	Implemented OMA-ARC-REST-NetAPI-2012-0083R01-CR_AudioCall_CONR_A0029_JSON
	12 Mar 2012	D.24	OMA-ARC-REST-NetAPI-2012-0096-CR_AC_small_fix implemented
	13 Mar 2012	Various	Editorial: Changed front cover, header and footer
Candidate Version OMA-TS-REST_NetAPI_AudioCall-V1_0	27 Mar 2012	n/a	Status changed to Candidate by TP TP Ref # OMA-TP-2012-0128-INP_REST_NetAPI_AudioCall_1_0_ERP_and_ETR_for_Candidate_Approval

Document Identifier	Date	Sections	Description
Draft Versions OMA-TS-REST_NetAPI_AudioCall-V1_0	06 Aug 2012	5.2.2.8, 5.2.2.9, 5.2.2.14, 5.2.2.15, 5.2.2.16, C.1, C.2, C.3, C.4, C.5, C.6, C.7	Incorporated CR: OMA-ARC-REST-NetAPI-2012-0208-CR_Issue_20_clientCorrelator_resolution_Audiocall Editorial changes
	27 Nov 2012	4.1, 6, G.1.2	Incorporated CR: OMA-ARC-REST-NetAPI-2012-0282-CR_AudioCall_implement_blueprint_longpoll_auth Template changed to OMA-TEMPLATE-TS_RESTful_Network_API-20120813-I Editorial changes
	08 Feb 2013	2.2	Reference to OMA Dictionary updated to version 2.9. Template updated.
Candidate Version OMA-TS-REST_NetAPI_AudioCall-V1_0	12 Feb 2013	n/a	Status changed to Candidate by TP TP Ref # OMA-TP-2013-0052-INP_REST_NetAPI_AudioCall_V1_0_Candidate_ERP_for_Notification

Appendix B. Static Conformance Requirements (Normative)

The notation used in this appendix is specified in [SCRRULES].

B.1 SCR for REST.AC Server

Item	Function	Reference	Requirement
REST-AC-SUPPORT-S-001-M	Support for the RESTful Audio Call API	5, 6	
REST-AC-SUPPORT-S-002-M	Support for the XML request & response format	6	
REST-AC-SUPPORT-S-003-M	Support for the JSON request & response format	6	
REST-AC-SUPPORT-S-004-O	Support for the application/x-www-form-urlencoded	Appendix C	

B.1.1 SCR for REST.AC.Messages Server

Item	Function	Reference	Requirement
REST-AC-MSG-S-001-O	Support for Audio Call Messages	6.1	REST-AC-MSG-S-002-O
REST-AC-MSG-S-002-O	Retrieving a list of all Audio Call Messages – GET	6.1.3	

B.1.2 SCR for REST.AC.Messages.Text Server

Item	Function	Reference	Requirement
REST-AC-MSG-TEXT-S-001-O	Support for Audio Call text messages	6.2	REST-AC-MSG-TEXT-S-003-O AND REST-AC-MSG-INDTEXT-S-001-O
REST-AC-MSG-TEXT-S-002-O	Retrieving a list of all Audio Call text messages – GET	6.2.3	
REST-AC-MSG-TEXT-S-003-O	Create a new Audio Call text message – POST (XML or JSON)	6.2.5	
REST-AC-MSG-TEXT-S-004-O	Create a new Audio Call text message – POST (application/x-www-form-urlencoded)	C.1	

B.1.3 SCR for REST.AC.Messages.IndividualText Server

Item	Function	Reference	Requirement
REST-AC-MSG-INDTEXT-S-001-O	Support for individual Audio Call text message	6.3	REST-AC-MSG-INDTEXT-S-002-

Item	Function	Reference	Requirement
			O AND REST-AC-MSG-INDTEXT-S-003-O AND REST-AC-MSG-INDTEXT-STAT-S-001-O
REST-AC-MSG-INDTEXT-S-002-O	Retrieving an individual Audio Call text message – GET	6.3.3	
REST-AC-MSG-INDTEXT-S-003-O	Terminate and remove an individual Audio Call text message – DELETE	6.3.6	

B.1.4 SCR for REST.AC.Messages.IndividualText.Status Server

Item	Function	Reference	Requirement
REST-AC-MSG-INDTEXT-STAT-S-001-O	Support for individual Audio Call text message status	6.4	REST-AC-MSG-INDTEXT-STAT-S-002-O
REST-AC-MSG-INDTEXT-STAT-S-002-O	Retrieving status for an individual Audio Call text message – GET	6.4.3	

B.1.5 SCR for REST.AC.Messages.Audio Server

Item	Function	Reference	Requirement
REST-AC-MSG-AUDIO-S-001-M	Support for Audio Call audio messages	6.5	
REST-AC-MSG-AUDIO-S-002-O	Retrieving all Audio Call audio messages – GET	6.5.3	
REST-AC-MSG-AUDIO-S-003-M	Create a new Audio Call audio message – POST (XML or JSON)	6.5.5	
REST-AC-MSG-AUDIO-S-004-O	Create a new Audio Call audio message – POST (application/x-www-form-urlencoded)	C.2	

B.1.6 SCR for REST.AC.Messages.IndividualAudio Server

Item	Function	Reference	Requirement
REST-AC-MSG-INDAUDIO-S-001-M	Support for individual Audio Call audio message	6.6	
REST-AC-MSG-INDAUDIO-S-002-M	Retrieving an individual Audio Call audio message – GET	6.6.3	
REST-AC-MSG-INDAUDIO-S-003-M	Terminate and remove an individual Audio Call audio message – DELETE	6.6.6	

B.1.7 SCR for REST.AC.Messages.IndividualAudio.Status Server

Item	Function	Reference	Requirement
REST-AC-MSG-INDAUDIO-STAT-S-001-M	Support for individual Audio Call audio message status	6.7	
REST-AC-MSG-INDAUDIO-STAT-S-002-M	Retrieving status for an individual Audio Call audio message – GET	6.7.3	

B.1.8 SCR for REST.AC.Messages.VXML Server

Item	Function	Reference	Requirement
REST-AC-MSG-VXML-S-001-O	Support for Audio Call Voice XML messages	6.8	REST-AC-MSG-VXML-S-003-O AND REST-AC-MSG-INDVXML-S-001-O
REST-AC-MSG-VXML-S-002-O	Retrieving all Audio Call Voice XML messages – GET	6.8.3	
REST-AC-MSG-VXML-S-003-O	Create a new Audio Call Voice XML message – POST (XML or JSON)	6.8.5	
REST-AC-MSG-VXML-S-004-O	Create a new Audio Call Voice XML message – POST (application/x-www-form-urlencoded)	C.3	

B.1.9 SCR for REST.AC.Messages.IndividualVXML Server

Item	Function	Reference	Requirement
REST-AC-MSG-INDVXML-S-001-O	Support for individual Audio Call Voice XML message	6.9	REST-AC-MSG-INDVXML-S-002-O AND REST-AC-MSG-INDVXML -S-003-O AND REST-AC-MSG-INDVXML-STAT-S-001-O
REST-AC-MSG-INDVXML-S-002-O	Retrieving an individual Audio Call Voice XML message – GET	6.9.3	
REST-AC-MSG-INDVXML-S-003-O	Terminate and remove an individual Audio Call Voice XML message – DELETE	6.9.6	

B.1.10 SCR for REST.AC.Messages.IndividualVXML.Status Server

Item	Function	Reference	Requirement
REST-AC-MSG-INDVXML-STAT-S-001-O	Support for individual Audio Call Voice XML message status	6.10	REST-AC-MSG-INDVXML-STAT-S-002-O
REST-AC-MSG-INDVXML-	Retrieving status for an individual	6.10.3	

Item	Function	Reference	Requirement
STAT-S-002-O	Audio Call audio message – GET		

B.1.11 SCR for REST.AC.Messages.Video Server

Item	Function	Reference	Requirement
REST-AC-MSG-VIDEO-S-001-O	Support for Audio Call video messages	6.11	REST-AC-MSG-VIDEO -S-003-O AND REST-AC-MSG-INDVIDEO-S-001-O
REST-AC-MSG-VIDEO-S-002-O	Retrieving all Audio Call video messages – GET	6.11.3	
REST-AC-MSG-VIDEO-S-003-O	Create a new Audio Call video message – POST (XML or JSON)	6.11.5	
REST-AC-MSG-VIDEO-S-004-O	Create a new Audio Call video message – POST (application/x-www-form-urlencoded)	C.4	

B.1.12 SCR for REST.AC.Messages.IndividualVideo Server

Item	Function	Reference	Requirement
REST-AC-MSG-INDVIDEO-S-001-O	Support for individual Audio Call video message	6.12	REST-AC-MSG-INDVIDEO-S-002-O AND REST-AC-MSG-INDVIDEO-S-003-O AND REST-AC-MSG-INDVIDEO-STAT-S-001-O
REST-AC-MSG-INDVIDEO-S-002-O	Retrieving an individual Audio Call video message – GET	6.12.3	
REST-AC- MSG-INDVIDEO-S-003-O	Terminate and remove an individual Audio Call video message – DELETE	6.12.6	

B.1.13 SCR for REST.AC.Messages.IndividualVideo.Status Server

Item	Function	Reference	Requirement
REST-AC-MSG-INDVIDEO-STAT-S-001-O	Support for individual Audio Call video message status	6.13	REST-AC-INDVIDEO-STAT-S-002-O
REST-AC-MSG-INDVIDEO-STAT-S-002-O	Retrieving status for an individual Audio Call video message – GET	6.13.3	

B.1.14 SCR for REST.AC.MediaInteracions Server

Item	Function	Reference	Requirement
REST-AC-MEDINT-S-001-O	Support for all Audio Call media capture interactions	6.14	REST-AC-MEDINT-S-002-O
REST-AC-MEDINT-S-002-O	Retrieving all active Audio Call media capture interactions – GET	6.14.3	

B.1.15 SCR for REST.AC.MediaInteractions.PlayAndCollect Server

Item	Function	Reference	Requirement
REST-AC-MEDINT-PAC-S-001-M	Support for Audio Call Play and Collect interactions	6.15	
REST-AC-MEDINT-PAC-S-002-O	Retrieving all Audio Call Play and Collect interaction – GET	6.15.3	
REST-AC-MEDINT-PAC-S-003-M	Create a new Audio Call Play and Collect interaction – POST (XML or JSON)	6.15.5	
REST-AC-MEDINT-PAC-S-004-O	Create a new Audio Call Play and Collect interaction – POST (application/x-www-form-urlencoded)	C.5	

B.1.16 SCR for REST.AC.MediaInteractions.IndividualPlayAndCollect Server

Item	Function	Reference	Requirement
REST-AC-MEDINT-INDPAC-S-001-M	Support for individual Audio Call Play and Collect interaction message	6.16	
REST-AC-MEDINT-INDPAC-S-002-M	Retrieving an individual Audio Call Play and Collect interaction – GET	6.16.3	
REST-AC-MEDINT-INDPAC-S-003-M	Terminate and remove an individual Audio Call Play and Collect interaction – DELETE	6.16.6	

B.1.17 SCR for REST.AC.MediaInteractions.PlayAndRecord Server

Item	Function	Reference	Requirement
REST-AC-MEDINT-PAR-S-001-O	Support for Audio Call Play and Record interactions	6.17	REST-AC-MEDINT-PAR-S-003-O AND REST-AC-MEDINT-INDPAR-S-001-O AND REST-AC-MEDINT-NOTIF-

Item	Function	Reference	Requirement
			S-002-O
REST-AC-MEDINT-PAR-S-002-O	Retrieving all Audio Call Play and Record interactions – GET	6.17.3	
REST-AC-MEDINT-PAR-S-003-O	Create a new Audio Call Play and Record interaction – POST (XML or JSON)	6.17.5	
REST-AC-MEDINT-PAR-S-004-O	Create a new Audio Call Play and Record interactions – POST (application/x-www-form-urlencoded)	C.6	

B.1.18 SCR for REST.AC.MediaInteractions.IndividualPlayAndRecord Server

Item	Function	Reference	Requirement
REST-AC-MEDINT-INDPAR-S-001-O	Support for individual Audio Call Play and Record interaction message	6.18	REST-AC-MEDINT-INDPAR-S-002-O AND REST-AC-MEDINT-INDPAR-S-003-O
REST-AC-MEDINT-INDPAR-S-002-O	Retrieving an individual Audio Call Play and Record interaction – GET	6.18.3	
REST-AC-MEDINT-INDPAR-S-003-O	Terminate and remove an individual Audio Call Play and Record interaction – DELETE	6.18.6	

B.1.19 B.1.19 SCR for REST.AC.MediaInteractions.Notification Server

Item	Function	Reference	Requirement
REST-AC-MEDINT-NOTIF-S-001-M	Support for notifications related to Audio Call Play and Collect interactions	[REST_NetAPI_CallNotif]	REST-CN-SUBSCR-PAC-S-001-M AND REST-CN-SUBSCR-PAC-S-003-M AND REST-CN-SUBSCR-INDPAC-S-001-M AND REST-CN-SUBSCR-INDPAC-S-002-

Item	Function	Reference	Requirement
			M AND REST-CN-SUBSCR-INDPAC-S-003-M AND REST-CN-NOTIF-MEDINT-S-001-M AND REST-CN-NOTIF-MEDINT-S-002-M
REST-AC-MEDINT-NOTIF-S-002-O	Support for notifications related to Audio Call Play and Record interactions	[REST_NetAPI_CallNotif]	REST-CN-SUBSCR-PAR-S-001-O AND REST-CN-NOTIF-MEDINT-S-001-M AND REST-CN-NOTIF-MEDINT-S-002-M
REST-AC-MEDINT-NOTIF-S-003-O	Support for notifications related to Audio Call speech recognition interactions	[REST_NetAPI_CallNotif]	REST-CN-SUBSCR-ASR-S-001-O AND REST-CN-NOTIF-MEDINT-S-001-M AND REST-CN-NOTIF-MEDINT-S-002-M

B.1.20 SCR for REST.AC.MediaInteractions.Recognition Server

Item	Function	Reference	Requirement
REST-AC-MEDINT-ASR-S-001-O	Support for Audio Call speech recognition interactions	6.19	REST-AC-MEDINT-ASR-S-003-O AND REST-AC-MEDINT-INDASR-S-001-O

Item	Function	Reference	Requirement
			AND REST-AC-MEDINT-NOTIF-S-003-O
REST-AC-MEDINT-ASR-S-002-O	Retrieving all Audio Call speech recognition interactions – GET	6.19.3	
REST-AC-MEDINT-ASR-S-003-O	Create a new Audio Call speech recognition interaction – POST (XML or JSON)	6.19.5	
REST-AC-MEDINT-ASR-S-004-O	Create a new Audio Call speech recognition interaction – POST (application/x-www-form-urlencoded)	C.7	

B.1.21 SCR for REST.AC.MediaInteractions.IndividualRecognition Server

Item	Function	Reference	Requirement
REST-AC-MEDINT-INDASR-S-001-O	Support for individual Audio Call speech recognition interaction message	6.20	REST-AC-MEDINT-INDASR-S-002-O AND REST-AC-MEDINT-INDASR-S-003-O
REST-AC-MEDINT-INDASR-S-002-O	Retrieving an individual Audio Call speech recognition interaction – GET	6.20.3	
REST-AC-MEDINT-INDASR-S-003-O	Terminate and remove an individual Audio Call speech recognition interaction – DELETE	6.20.6	

Appendix C. Application/x-www-form-urlencoded Request Format for POST Operations (Normative)

This section defines a format for RESTful Audio Call API requests where the body of the request is encoded using the application/x-www-form-urlencoded MIME type.

Note: only the request body is encoded as application/x-www-form-urlencoded, the response is still encoded as XML or JSON depending on the preference of the client and the capabilities of the server.

Names and values MUST follow the application/x-www-form-urlencoded character escaping rules at [W3C_URLENC].

The encoding is defined below for all Audio Call REST operations which are based on POST requests.

C.1 Play text message to call participants

This operation is used to create a new text message to be played to call participants, see section 6.2.5.

The request parameters are as follows:

Name	Type/Values	Optional	Description
callSessionIdentifier	xsd:string	No	Identifies the call session to which the message is played.
callParticipant	xsd:anyURI [0..unbounded]	Yes	The set of participant addresses(e.g. 'sip' URI, 'tel' URI, 'acr' URI) contained within the callSession to which the message is to be played. If no participants are specified, the message is played to all participants.
text	xsd:string	No	Text to process with a Text-To-Speech engine.
lang	xsd:string	Yes	Language of text. The format of this parameter is aligned with that of the built-in XML attribute xml:lang [W3C-XML11]. It is recommended to provide this parameter.
chargingDescription	xsd:string [0..unbounded]	Yes	Description of charge to apply to this message. In case charging is required, this parameter MUST be present.
chargingCurrency	xsd:string	Yes	Currency of charge to apply to this message. In case chargingDescription is not present, this parameter MUST NOT be present.
chargingAmount	xsd:decimal	Yes	Charging amount to apply to this message. In case chargingDescription is not present, this parameter MUST NOT be present.
chargingCode	xsd:string	Yes	Charging code to apply to this message. In case chargingDescription is not present, this parameter MUST NOT be present.

clientCorrelator	xsd:string	Yes	A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server. This element SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids playing the same message twice in such situations. In case the element is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it.

If the operation was successful, it returns an HTTP Status of “201 Created”.

C.1.1 Example (Informative)

C.1.1.1 Request

```
POST /exampleAPI/audiocall/v1/messages/text HTTP/1.1
Accept: application/xml
Content-Length: nnnn
Content-Type: application/x-www-form-urlencoded
Host: example.com

callSessionIdentifier=A45678&
callParticipant=tel%3A%2B19585550101&
callParticipant=tel%3A%2B19585550102&
text=Welcome%20to%20the%20telephone%20conference&
lang=en&
clientCorrelator=12345
```

C.1.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/messages/text/msg123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:textMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
      <callSessionIdentifier>A45678</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <callParticipant>tel:+19585550102</callParticipant>
      <text xml:lang="en">Welcome to the telephone conference</text>
      <messageStatusList>
         <messageStatus>
            <callParticipant>tel:+19585550101</callParticipant>
            <status>Pending</status>
```

```
        </messageStatus>
        <messageStatus>
          <callParticipant>tel:+19585550102</callParticipant>
          <status>Pending</status>
        </messageStatus>
        <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList</resourceURL>
      </messageStatusList>
      <clientCorrelator>12345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/text/msg123</resourceURL>
    </ac:textMessage>
```

C.2 Play audio message to call participants

This operation is used to create a new audio message to be played to call participants, see section 6.5.5.

The request parameters are as follows:

Name	Type/Values	Optional	Description
callSessionIdentifier	xsd:string	No	Identifies the call session to which the message is played.
callParticipant	xsd:anyURI [0..unbounded]	Yes	The set of participant addresses (e.g. 'sip' URI, 'tel' URI, 'acr' URI) contained within the callSession to which the message is to be played. If no participants are specified, the message is played to all participants.
mediaUrl	xsd:anyURI	No	Location of content (audio, video, voiceXml) to play.
mediaType	xsd:string	Yes	MIME media type of the content to be played.
chargingDescription	xsd:string [0..unbounded]	Yes	Description of charge to apply to this message. In case charging is required, this parameter MUST be present.
chargingCurrency	xsd:string	Yes	Currency of charge to apply to this message. In case chargingDescription is not present, this parameter MUST NOT be present.
chargingAmount	xsd:decimal	Yes	Charging amount to apply to this message. In case chargingDescription is not present, this parameter MUST NOT be present.
chargingCode	xsd:string	Yes	Charging code to apply to this message. In case chargingDescription is not present, this parameter MUST NOT be present.

clientCorrelator	xsd:string	Yes	A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server. This element SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids playing the same message twice in such situations. In case the element is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it.

If the operation was successful, it returns an HTTP Status of “201 Created”.

C.2.1 Example 1, using tel URI (Informative)

C.2.1.1 Request

```
POST /exampleAPI/audiocall/v1/messages/audio HTTP/1.1
Accept: application/xml
Content-Length: nnnn
Content-Type: application/x-www-form-urlencoded
Host: example.com

callSessionIdentifier=B45678&
callParticipant= tel%3A%2B19585550101&
callParticipant= tel%3A%2B19585550102&
mediaUrl=http%3A%2F%2Fwww.example.com%2Fann1.mp3&
mediaType=audio%2Fmpeg&
clientCorrelator=22345
```

C.2.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:audioMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
<callSessionIdentifier>B45678</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <callParticipant>tel:+19585550102</callParticipant>
       <mediaUrl>http://www.example.com/ann1.mp3</mediaUrl>
       <mediaType>audio/mpeg</mediaType>
        <messageStatusList>
          <messageStatus>
             <callParticipant>tel:+19585550101</callParticipant>
```



```
            <status>Pending</status>
         </messageStatus>
         <messageStatus>
            <callParticipant>tel:+19585550102</callParticipant>
            <status>Pending</status>
         </messageStatus>
         <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList</resourceURL>
      </messageStatusList>
      <clientCorrelator>22345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123</resourceURL>
</ac:audioMessage>
```

C.2.2 Example 2, using ACR (Informative)

C.2.2.1 Request

```
POST /exampleAPI/audiocall/v1/messages/audio HTTP/1.1
Accept: application/xml
Content-Length: nnnn
Content-Type: application/x-www-form-urlencoded
Host: example.com

callSessionIdentifier=B45678&
callParticipant= acr%3Apseudonym123&
callParticipant= acr%3Apseudonym456&
mediaUrl=http%3A%2F%2Fwww.example.com%2Fann1.mp3&
mediaType=audio%2Fmpeg&
clientCorrelator=22345
```

C.2.2.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:audioMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
<callSessionIdentifier>B45678</callSessionIdentifier>
      <callParticipant>acr:pseudonym123</callParticipant>
      <callParticipant>acr:pseudonym456</callParticipant>
       <mediaUrl>http://www.example.com/ann1.mp3</mediaUrl>
       <mediaType>audio/mpeg</mediaType>
        <messageStatusList>
          <messageStatus>
            <callParticipant>acr:pseudonym123</callParticipant>
            <status>Pending</status>
          </messageStatus>
          <messageStatus>
```



```
            <callParticipant>acr:pseudonym456</callParticipant>
            <status>Pending</status>
          </messageStatus>
          <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList</resourceURL>
       </messageStatusList>
      <clientCorrelator>22345</clientCorrelator>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123</resourceURL>
 </ac:audioMessage>
```

C.3 Play VoiceXML message to call participants

This operation is used to create a new VoiceXML message to be played to call participants, see section 6.8.5.

The request parameters are as follows:

Name	Type/Values	Optional	Description
callSessionIdentifier	xsd:string	No	Identifies the call session to which the message is played.
callParticipant	xsd:anyURI [0..unbounded]	Yes	The set of participant (e.g. 'sip' URI, 'tel' URI, 'acr' URI) addresses contained within the callSession to which the message is to be played. If no participants are specified, the message is played to all participants.
mediaUrl	xsd:anyURI	No	Location of content (audio, video, voiceXml) to play.
mediaType	xsd:string	Yes	MIME media type of the content to be played.
chargingDescription	xsd:string [0..unbounded]	Yes	Description of charge to apply to this message. In case charging is required, this parameter MUST be present.
chargingCurrency	xsd:string	Yes	Currency of charge to apply to this message. In case chargingDescription is not present, this parameter MUST NOT be present.
chargingAmount	xsd:decimal	Yes	Charging amount to apply to this message. In case chargingDescription is not present, this parameter MUST NOT be present.
chargingCode	xsd:string	Yes	Charging code to apply to this message. In case chargingDescription is not present, this parameter MUST NOT be present.

clientCorrelator	xsd:string	Yes	A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server. This element SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids playing the same message twice in such situations. In case the element is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it.

If the operation was successful, it returns an HTTP Status of "201 Created".

C.3.1 Example (Informative)

C.3.1.1 Request

```
POST /exampleAPI/audiocall/v1/messages/voiceXml HTTP/1.1
Accept: application/xml
Content-Length: nnnn
Content-Type: application/x-www-form-urlencoded
Host: example.com

callSessionIdentifier=D45678&
callParticipant= tel%3A%2B19585550101&
callParticipant= tel%3A%2B19585550102&
mediaUrl=http%3A%2F%2Fwww.example.com%2Fann1.vxml&
clientCorrelator=42345
```

C.3.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:voiceXMLMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
      <callSessionIdentifier>D45678</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <callParticipant>tel:+19585550102</callParticipant>
       <mediaUrl>http://www.example.com/ann1.vxml</mediaUrl>
         <messageStatusList>
           <messageStatus>
              <callParticipant>tel:+19585550101</callParticipant>
              <status>Pending</status>
           </messageStatus>
```

```
        <messageStatus>
           <callParticipant>tel:+19585550102</callParticipant>
           <status>Pending</status>
        </messageStatus>
       <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList</resourceURL>
    </messageStatusList>
   <clientCorrelator>42345</clientCorrelator>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123</resourceURL>
</ac:voiceXMLMessage>
```

C.4 Play video message to call participants

This operation is used to create a new video message to be played to call participants, see section 6.11.5.

The request parameters are as follows:

Name	Type/Values	Optional	Description
callSessionIdentifier	xsd:string	No	Identifies the call session to which the message is played.
callParticipant	xsd:anyURI [0..unbounded]	Yes	The set of participant addresses (e.g. 'sip' URI, 'tel' URI, 'acr' URI) contained within the callSession to which the message is to be played. If no participants are specified, the message is played to all participants.
mediaUrl	xsd:anyURI	No	Location of content (audio, video, voiceXml) to play.
mediaType	xsd:string	Yes	MIME media type of the content to be played.
chargingDescription	xsd:string [0..unbounded]	Yes	Description of charge to apply to this message. In case charging is required, this parameter MUST be present.
chargingCurrency	xsd:string	Yes	Currency of charge to apply to this message. In case chargingDescription is not present, this parameter MUST NOT be present.
chargingAmount	xsd:decimal	Yes	Charging amount to apply to this message. In case chargingDescription is not present, this parameter MUST NOT be present.
chargingCode	xsd:string	Yes	Charging code to apply to this message. In case chargingDescription is not present, this parameter MUST NOT be present.

clientCorrelator	xsd:string	Yes	A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server. This element SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids playing the same message twice in such situations. In case the element is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it.

If the operation was successful, it returns an HTTP Status of "201 Created".

C.4.1 Example (Informative)

C.4.1.1 Request

```
POST /exampleAPI/audiocall/v1/messages/video HTTP/1.1
Accept: application/xml
Content-Length: nnnn
Content-Type: application/x-www-form-urlencoded
Host: example.com

callSessionIdentifier=E45678&
callParticipant= tel%3A%2B19585550101&
callParticipant= tel%3A%2B19585550102&
mediaUrl=http%3A%2F%2Fwww.example.com%2Fann1.mp4&
mediaType=video%2Fmp4&
clientCorrelator=52345
```

C.4.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/messages/video/msg123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:videoMessage xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
      <callSessionIdentifier>E45678</callSessionIdentifier>
      <callParticipant>tel:+19585550101</callParticipant>
      <callParticipant>tel:+19585550102</callParticipant>
       <mediaUrl>http://www.example.com/ann1.mp4</mediaUrl>
       <mediaType>video/mp4</mediaType>
        <messageStatusList>
          <messageStatus>
             <callParticipant>tel:+19585550101</callParticipant>
             <status>Pending</status>
```



```
        </messageStatus>
        <messageStatus>
          <callParticipant>tel:+19585550102</callParticipant>
          <status>Pending</status>
        </messageStatus>
      <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList</resourceURL>
    </messageStatusList>
    <clientCorrelator>52345</clientCorrelator>
    <resourceURL>http://example.com/exampleAPI/audiocall/v1/messages/video/msg123</resourceURL>
</ac:videoMessage>
```

C.5 Play a media file and collect digits

This operation is used to play a media file to call participants and collect digits from the participants, see section 6.15.5.

The request parameters are as follows:

Name	Type/Values	Optional	Description
callSessionIdentifier	xsd:string	No	Identifies the call session to which the message is played.
callParticipant	xsd:anyURI [0..unbounded]	Yes	The set of participant addresses (e.g. 'sip' URI, 'tel' URI, 'acr' URI) contained within the callSession to which the message is to be played. If no participants are specified, the message is played to all participants.
playFileLocation	xsd:anyURI	Choice	The location of the file that will be played to the endpoint, including VoiceXML script location.
textString	xsd:string	Choice	The text to be converted by a Text-To-Speech engine.
messageFormat	AnnouncementFormat	No	The type of announcement prompt to play to the end user.
mediaType	xsd:string	Yes	MIME media type of the content to be played.
interruptMedia	xsd:boolean	No	Indicates whether the application allows the end user to interrupt, or pause, the prompt.
minDigits	xsd:unsignedInt	Yes	The minimum number of digits that will be collected. If this isn't achieved, then a default prompt shall be played requesting for more digits to be entered. If not given, the behaviour is implementation-specific.
maxDigits	xsd:unsignedInt	Yes	The maximum number of digits that will be collected. If not given, the behaviour is implementation-specific.

endChar	xsd:string	Yes	If the user enters via the phone keypad the character specified in this element, this indicates to stop collecting digits. This string SHALL have a minimum and maximum length of one; valid characters are the digits '0' to '9', asterisk ('*') and hash ('#'). If pressed, the end character SHALL neither be returned nor count towards the number of digits to be collected as defined by minDigits and maxDigits. Note: Without this the user will have to wait for a timeout after entering his/her digits. Note: In case endChar is not given, the behavior (i.e. whether or not a particular character such as '#' triggers the end of the digit collection) is network-specific.
clientCorrelator	xsd:string	Yes	A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server. This element SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids starting the digit capture twice in such situations. In case the element is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it.

If the operation was successful, it returns an HTTP Status of "201 Created".

Either playFileLocation or textString but not both SHALL be provided.

C.5.1 Example (Informative)

C.5.1.1 Request

```
POST /exampleAPI/audiocall/v1/interactions/collection HTTP/1.1
Accept: application/xml
Content-Length: nnnn
Content-Type: application/x-www-form-urlencoded
Host: example.com

callSessionIdentifier=F14567&
callParticipant= tel%3A%2B19585550101&
playFileLocation=http%3A%2F%2Fwww.example.com%2Fmsg1.mp3&
messageFormat=Audio&
mediaType=audio%2Fmpeg&
```



```
interruptMedia=true&
minDigits=1&
maxDigits=1&
endChar=%23&
clientCorrelator=62345
```

C.5.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:digitCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>F14567</callSessionIdentifier>
   <callParticipant>tel:+19585550101</callParticipant>
   <playingConfiguration>
      <playFileLocation>http://www.example.com/msg1.mp3</playFileLocation>
      <messageFormat>Audio</messageFormat>
      <mediaType>audio/mpeg</mediaType>
      <interruptMedia>true</interruptMedia>
   </playingConfiguration>
   <digitConfiguration>
      <minDigits>1</minDigits>
      <maxDigits>1</maxDigits>
      <endChar>#</endChar>
   </digitConfiguration>
   <clientCorrelator>62345</clientCorrelator>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123</resourceURL>
</ac:digitCapture>
```

C.6 Play a media file and record interaction

This operation is used to play a media file to call participants and record a response from the participants, see section 6.17.5.

The request parameters are as follows:

Name	Type/Values	Optional	Description
callSessionIdentifier	xsd:string	No	Identifies the call session to which the message is played.
callParticipant	xsd:anyURI [0..unbounded]	Yes	The set of participant addresses (e.g. 'sip' URI, 'tel' URI, 'acr' URI) contained within the callSession to which the message is to be played. If no participants are specified, the message is played to all participants.
playFileLocation	xsd:anyURI	Choice	The location of the file that will be played to the endpoint, including VoiceXML script location.

textString	xsd:string	Choice	The text to be converted by a Text-To-Speech engine.
messageFormat	AnnouncementFormat	No	The type of announcement prompt to play to the end user.
mediaType	xsd:string	Yes	MIME media type of the content to be played.
interruptMedia	xsd:boolean	No	Indicates whether the application allows the end user to interrupt, or pause, the prompt.
recFileLocation	xsd:anyURI	Yes	The location for storing the information recorded from the terminal. If not given, the behaviour is implementation-specific.
maxRecordingLength	xsd:int	Yes	Maximum length in seconds of recording to be captured.
endChar	xsd:string	Yes	If the user enters via the phone keypad the character specified in this element, this indicates to stop recording. This string SHALL have a minimum and maximum length of one; valid characters are the digits '0' to '9', asterisk ('*') and hash ('#'). Note: Without this the user will have to wait for a timeout after entering his/her digits. Note: In case endChar is not given, the behavior (i.e. whether or not a particular character such as '#' triggers the end of the recording) is network-specific.
clientCorrelator	xsd:string	Yes	A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server. This element SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids starting the digit capture twice in such situations. In case the element is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it.

If the operation was successful, it returns an HTTP Status of "201 Created".

Either playFileLocation or textString but not both SHALL be provided.

C.6.1 Example (Informative)

C.6.1.1 Request

```
POST /exampleAPI/audiocall/v1/interactions/recording HTTP/1.1
Accept: application/xml
```

```
Content-Length: nnnn
Content-Type: application/x-www-form-urlencoded
Host: example.com

callSessionIdentifier=F34567&
callParticipant= tel%3A%2B19585550102&
playFileLocation=http%3A%2F%2Fwww.example.com%2Fmsg2.mp3&
messageFormat=Audio&
mediaType=audio%2Fmpeg&
interruptMedia=true&
recFileLocation=http%3A%2F%2Fwww.example.com%2Frec2.mp3&
maxRecordingLength=10&
endChar=%23&
clientCorrelator=82345
```

C.6.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/interactions/recording/int123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:recordingCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
   <callSessionIdentifier>F34567</callSessionIdentifier>
   <callParticipant>tel:+19585550102</callParticipant>
   <playingConfiguration>
      <playFileLocation>http://www.example.com/msg2.mp3</playFileLocation>
      <messageFormat>Audio</messageFormat>
      <mediaType>audio/mpeg</mediaType>
      <interruptMedia>true</interruptMedia>
   </playingConfiguration>
   <recordingConfiguration>
      <recFileLocation>http://www.example.com/rec2.mp3</recFileLocation>
      <maxRecordingLength>10</maxRecordingLength>
      <endChar>#</endChar>
   </recordingConfiguration>
   <clientCorrelator>82345</clientCorrelator>
   <resourceURL>http://example.com/exampleAPI/audiocall/v1/interactions/recording/int123</resourceURL>
</ac:recordingCapture>
```

C.7 Engage participants in a dialog and recognize their spoken response

This resource is used to engage the participants in a dialog and automatically recognize their spoken responses, see section 6.19.5.

The request parameters are as follows:

Name	Type/Values	Optional	Description
callSessionIdentifier	xsd:string	Choice	Identifies the call session for the media interaction.
link	xsd:anyURI	Choice	Identifies (by a REST resource URL) the call session for the media interaction. The link MUST point to a resource of type "CallSessionInformation".
callParticipant	xsd:anyURI [0..unbounded]	Yes	The set of participant addresses (e.g. 'sip' URI, 'tel' URI, 'acr' URI) contained within the call session with whom to interact. If no participants are specified, this applies to all participants in the call session.
scriptFileLocation	xsd:anyURI	No	The location of the script file, e.g. VoiceXML. The meaning of this URI is implementation specific. Note that usually, announcement media are pre-loaded into specific network nodes, but some implementations might also allow playing content available on internet servers.
scriptType	xsd:string	No	The MIME type of the script.
endChar	xsd:string	Yes	If the user enters via the phone keypad the character specified in this element, this indicates to stop the dialog. This string SHALL have a minimum and maximum length of one; valid characters are the digits '0' to '9', asterisk ('*') and hash ('#'). Note: Without this the user will have to wait for a timeout after entering his/her digits. Note: In case endChar is not given, the behavior (i.e. whether or not a particular character such as '#' triggers the end of the recording) is network-specific.
clientCorrelator	xsd:string	Yes	A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server. This element SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids starting the speech recognition dialog twice in such situations. In case the element is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it.

resourceURL	xsd:anyURI	Yes	Self referring URL. The resourceURL SHALL NOT be included in POST requests by the client, but MUST be included in POST requests representing notifications by the server to the client, when a complete representation of the resource is embedded in the notification. The resourceURL MUST also be included in responses to any HTTP method that returns an entity body, and in PUT requests.

If the operation was successful, it returns an HTTP Status of “201 Created”.

Either scriptFileLocation or script but not both SHALL be provided. Further, either callSessionIdentifier or link SHALL be provided.

C.7.1 Example (Informative)

C.7.1.1 Request

```
POST /exampleAPI/audiocall/v1/interactions/recognition HTTP/1.1
Accept: application/xml
Content-Length: nnnn
Content-Type: application/x-www-form-urlencoded
Host: example.com

callSessionIdentifier=G34567&
callParticipant=tel%3A%2B19585550102&
scriptFileLocation>http%3A%2F%2Fwww.example.com%2Fasr.vxml&
scriptType=application%2Fvoicexml%2Bxml&
endChar=%23&
clientCorrelator=92345
```

C.7.1.2 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

<?xml version="1.0" encoding="UTF-8"?>
<ac:recognitionCapture xmlns:ac="urn:oma:xml:rest:netapi:audiocall:1">
    <callSessionIdentifier>G34567</callSessionIdentifier>
    <callParticipant>tel:+19585550102</callParticipant>
    <asrConfiguration>
        <scriptFileLocation>http://www.example.com/asr.vxml</scriptFileLocation>
        <scriptType>application/voicexml+xml</scriptType>
        <endChar>#</endChar>
    </asrConfiguration>
    <clientCorrelator>92345</clientCorrelator>
</ac:recognitionCapture>
```

Appendix D. JSON examples (Informative)

JSON (JavaScript Object Notation) is a lightweight, text-based, language-independent data interchange format. It provides a simple means to represent basic name-value pairs, arrays and objects. JSON is relatively trivial to parse and evaluate using standard JavaScript libraries, and hence is suited for invocations from browsers or other processors with JavaScript engines. Further information on JSON can be found at [RFC4627].

The following examples show the request or and response for various operations using a JSON binding. The examples follow the XML to JSON serialization guidelines rules in [REST_NetAPI_Common]. A JSON response may can be obtained by following using the content type negotiation guidelines mechanism specified in [REST_NetAPI_Common].

For full details on the operations themselves please refer to the section number indicated.

D.1 Retrieving a list of all active audio call messages (section 6.1.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/messages HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT


{"messageList": {
   "audioMessage": [
      {
         "callParticipant": [
            "tel:+19585550101",
            "tel:+19585550102"
         ],
         "callSessionIdentifier": "B45678",
         "clientCorrelator": "22345",
         "mediaType": "audio/mpeg",
         "mediaUrl": "http://www.example.com/ann1.mp3",
         "messageStatusList": {
            "messageStatus": [
               {
                  "callParticipant": "tel:+19585550101",
                  "status": "Played"
               },
               {
                  "callParticipant": "tel:+19585550102",
                  "status": "Pending"
               }
            ],
            "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList"
         },
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123"
```

```
    },
    {
      "callParticipant": [
        "tel:+19585550103",
        "tel:+19585550104"
      ],
      "callSessionIdentifier": "C45678",
      "clientCorrelator": "32345",
      "mediaType": "audio/mpeg",
      "mediaUrl": "http://www.example.com/ann2.mp3",
      "messageStatusList": {
        "messageStatus": [
          {
            "callParticipant": "tel:+19585550103",
            "status": "Played"
          },
          {
            "callParticipant": "tel:+19585550104",
            "status": "Playing"
          }
        ],
        "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg456/statusList"
      },
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg456"
    }
  ],
  "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages",
  "textMessage": {
    "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
    ],
    "callSessionIdentifier": "A45678",
    "clientCorrelator": "12345",
    "messageStatusList": {
      "messageStatus": [
        {
          "callParticipant": "tel:+19585550101",
          "status": "Played"
        },
        {
          "callParticipant": "tel:+19585550102",
          "status": "Pending"
        }
      ],
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList"
    },
    "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/text/msg123",
    "text": {
      "$t": "Welcome to the telephone conference",
      "lang": "en"
    }
  },
  "videoMessage": {
    "callParticipant": [
      "tel:+19585550101",
```



```
            "tel:+19585550102"
         ],
         "callSessionIdentifier": "E45678",
         "clientCorrelator": "52345",
         "mediaType": "video/mp4",
         "mediaUrl": "http://www.example.com/ann1.mp4",
         "messageStatusList": {
            "messageStatus": [
               {
                  "callParticipant": "tel:+19585550101",
                  "status": "Played"
               },
               {
                  "callParticipant": "tel:+19585550102",
                  "status": "Pending"
               }
            ],
            "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList"
         },
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/video/msg123"
      },
      "voiceXmlMessage": {
         "callParticipant": [
            "tel:+19585550101",
            "tel:+19585550102"
         ],
         "callSessionIdentifier": "D45678",
         "clientCorrelator": "42345",
         "mediaUrl": "http://www.example.com/ann1.vxml",
         "messageStatusList": {
            "messageStatus": [
               {
                  "callParticipant": "tel:+19585550101",
                  "status": "Played"
               },
               {
                  "callParticipant": "tel:+19585550102",
                  "status": "Pending"
               }
            ],
            "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList"
         },
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123"
      }
}}
```


D.2 Retrieving a list of all active audio call text messages (section 6.2.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/messages/text HTTP/1.1
Accept: application/json
Host: example.com
```


Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"messageList": {
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/text",
   "textMessage": {
      "callParticipant": [
         "tel:+19585550101",
         "tel:+19585550102"
      ],
      "callSessionIdentifier": "A45678",
      "clientCorrelator": "12345",
      "messageStatusList": {
         "messageStatus": [
            {
               "callParticipant": "tel:+19585550101",
               "status": "Played"
            },
            {
               "callParticipant": "tel:+19585550102",
               "status": "Pending"
            }
         ],
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList"
      },
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/text/msg123",
      "text": {
         "$t": " Welcome to the telephone conference",
         "lang": "en"
      }
   }
}}
```

D.3 Creating an audio call text message, response with copy of created resource (section 6.2.5.1)

Request:

```
POST /exampleAPI/audiocall/v1/messages/text HTTP/1.1
Accept: application/json
Content-Length: nnnn
Content-Type: application/json
Host: example.com

{"textMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
```

```
    ],
    "callSessionIdentifier": "A45678",
    "clientCorrelator": "12345",
    "text": {
        "$t": "Welcome to the telephone conference",
        "lang": "en"
    }
}}
```

Response:

```
HTTP/1.1 201 Created
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"textMessage": {
    "callParticipant": [
        "tel:+19585550101",
        "tel:+19585550102"
    ],
    "callSessionIdentifier": "A45678",
    "clientCorrelator": "12345",
    "messageStatusList": {
        "messageStatus": [
            {
                "callParticipant": "tel:+19585550101",
                "status": "Pending"
            },
            {
                "callParticipant": "tel:+19585550102",
                "status": "Pending"
            }
        ],
        "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList"
    },
    "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/text/msg123",
    "text": {
        "$t": "Welcome to the telephone conference",
        "lang": "en"
    }
}}
```

D.4 Creating an audio call text message, response with location of created resource (section 6.2.5.2)

Request:

```
POST /exampleAPI/audiocall/v1/messages/text HTTP/1.1
```

```
Accept: application/json
Content-Length: nnnn
Content-Type: application/json
Host: example.com

{"textMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "A45678",
   "clientCorrelator": "12345",
   "text": {
      "$t": "Welcome to the telephone conference",
      "lang": "en"
   }
}}
```

Response:

```
HTTP/1.1 201 Created
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"resourceReference": {"resourceURL": " http://example.com/exampleAPI/audiocall/v1/messages/text/msg123"}}
```

D.5 Retrieving an active audio call text message (section 6.3.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/messages/text/msg123?resFormat=JSON HTTP/1.1
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"textMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "A45678",
   "clientCorrelator": "12345",
   "messageStatusList": {
```

```
      "messageStatus": [
         {
            "callParticipant": "tel:+19585550101",
            "status": "Played"
         },
         {
            "callParticipant": "tel:+19585550102",
            "status": "Pending"
         }
      ],
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/text/msg123",
   "text": {
      "$t": "Welcome to the telephone conference",
      "lang": "en"
   }
}}
```

D.6 Terminating an active audio call text message (section 6.3.6.1)

Request:

```
DELETE /exampleAPI/audiocall/v1/messages/text/msg123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"textMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "A45678",
   "clientCorrelator": "12345",
   "messageStatusList": {
      "messageStatus": [
         {
            "callParticipant": "tel:+19585550101",
            "status": "Played"
         },
         {
            "callParticipant": "tel:+19585550102",
```

```
            "status": "Terminated"
          }
      ],
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/text/msg123",
   "text": {
      "$t": "Welcome to the telephone conference",
      "lang": "en"
   }
}}
```

D.7 Retrieving status of an active audio call text message (section 6.4.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/messages/text/msg123/statusList HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"messageStatusList": {
   "messageStatus": [
      {
         "callParticipant": "tel:+19585550101",
         "status": "Played"
      },
      {
         "callParticipant": "tel:+19585550102",
         "status": "Pending"
      }
   ],
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/text/msg123/statusList"
}}
```

D.8 Retrieving a list of all active audio call audio messages (section 6.5.3.1)

Request:


```
GET /exampleAPI/audiocall/v1/messages/audio HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"messageList": {
   "audioMessage": [
      {
         "callParticipant": [
            "tel:+19585550101",
            "tel:+19585550102"
         ],
         "callSessionIdentifier": "B45678",
         "clientCorrelator": "22345",
         "mediaType": "audio/mpeg",
         "mediaUrl": "http://www.example.com/ann1.mp3",
         "messageStatusList": {
            "messageStatus": [
               {
                  "callParticipant": "tel:+19585550101",
                  "status": "Played"
               },
               {
                  "callParticipant": "tel:+19585550102",
                  "status": "Pending"
               }
            ],
            "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList"
         },
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123"
      },
      {
         "callParticipant": [
            "tel:+19585550103",
            "tel:+19585550104"
         ],
         "callSessionIdentifier": "C45678",
         "clientCorrelator": "32345",
         "mediaType": "audio/mpeg",
         "mediaUrl": "http://www.example.com/ann2.mp3",
         "messageStatusList": {
            "messageStatus": [
               {
                  "callParticipant": "tel:+19585550103",
                  "status": "Played"
               },
               {
                  "callParticipant": "tel:+19585550104",
                  "status": "Playing"
               }
```

```
            ],
            "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg456/statusList"
         },
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg456"
      }
   ],
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio"
}}
```


D.9 Creating an audio call audio message, using tel URI (section 6.5.5.1)

Request:


```
POST /exampleAPI/audiocall/v1/messages/audio HTTP/1.1
Accept: application/json
Content-Length: nnnn
Content-Type: application/json
Host: example.com

{"audioMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "B45678",
   "clientCorrelator": "22345",
   "mediaType": "audio/mpeg",
   "mediaUrl": "http://www.example.com/ann1.mp3"
}}
```


Response:


```
HTTP/1.1 201 Created
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"audioMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "B45678",
   "clientCorrelator": "22345",
   "mediaType": "audio/mpeg",
   "mediaUrl": "http://www.example.com/ann1.mp3",
   "messageStatusList": {
      "messageStatus": [
```



```
            {
               "callParticipant": "tel:+19585550101",
               "status": "Pending"
            },
            {
               "callParticipant": "tel:+19585550102",
               "status": "Pending"
            }
         ],
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList"
      },
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123"
}}
```


D.10 Creating an audio call audio message, using ACR (section 6.5.5.2)

Request:


```
POST /exampleAPI/audiocall/v1/messages/audio HTTP/1.1
Accept: application/json
Content-Length: nnnn
Content-Type: application/json
Host: example.com

{"audioMessage": {
   "callParticipant": [
      "acr:pseudonym123",
      "acr:pseudonym456"
   ],
   "callSessionIdentifier": "B45678",
   "clientCorrelator": "22345",
   "mediaType": "audio/mpeg",
   "mediaUrl": "http://www.example.com/ann1.mp3"
}}
```


Response:


```
HTTP/1.1 201 Created
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"audioMessage": {
   "callParticipant": [
      "acr:pseudonym123",
      "acr:pseudonym456"
   ],
   "callSessionIdentifier": "B45678",
   "clientCorrelator": "22345",
```



```
   "mediaType": "audio/mpeg",
   "mediaUrl": "http://www.example.com/ann1.mp3",
   "messageStatusList": {
      "messageStatus": [
         {
            "callParticipant": "acr:pseudonym123",
            "status": "Pending"
         },
         {
            "callParticipant": "acr:pseudonym456",
            "status": "Pending"
         }
      ],
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123"
}}
```


D.11 Retrieving an active audio call audio message (section 6.6.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/messages/audio/msg123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response:


```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"audioMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "B45678",
   "clientCorrelator": "22345",
   "mediaType": "audio/mpeg",
   "mediaUrl": "http://www.example.com/ann1.mp3",
   "messageStatusList": {
      "messageStatus": [
         {
            "callParticipant": "tel:+19585550101",
            "status": "Played"
         },
         {
            "callParticipant": "tel:+19585550102",
```



```
            "status": "Pending"
          }
       ],
       "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList"
    },
    "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123"
}}
```


D.12 Terminating an active audio call audio message (section 6.6.6.1)

Request:

```
DELETE /exampleAPI/audiocall/v1/messages/audio/msg123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response:


```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"audioMessage": {
    "callParticipant": [
       "tel:+19585550101",
       "tel:+19585550102"
    ],
    "callSessionIdentifier": "B45678",
    "clientCorrelator": "22345",
    "mediaType": "audio/mpeg",
    "mediaUrl": "http://www.example.com/ann1.mp3",
    "messageStatusList": {
       "messageStatus": [
          {
             "callParticipant": "tel:+19585550101",
             "status": "Played"
          },
          {
             "callParticipant": "tel:+19585550102",
             "status": "Terminated"
          }
       ],
       "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList"
    },
    "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123"
}}
```


D.13 Retrieving an active audio call audio message status (section 6.7.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/messages/audio/msg123/statusList HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"messageStatusList": {
   "messageStatus": [
      {
         "callParticipant": "tel:+19585550101",
         "status": "Played"
      },
      {
         "callParticipant": "tel:+19585550102",
         "status": "Pending"
      }
   ],
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/audio/msg123/statusList"
}}
```

D.14 Retrieving a list of all active audio call VoiceXML messages (section 6.8.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/messages/voiceXml HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"messageList": {
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/voiceXml",
   "voiceXmlMessage": {
      "callParticipant": [
         "tel:+19585550101",
         "tel:+19585550102"
```

```
      ],
      "callSessionIdentifier": "B45678",
      "clientCorrelator": "42345",
      "mediaUrl": "http://www.example.com/ann1.vxml",
      "messageStatusList": {
         "messageStatus": [
            {
               "callParticipant": "tel:+19585550101",
               "status": "Played"
            },
            {
               "callParticipant": "tel:+19585550102",
               "status": "Pending"
            }
         ],
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList"
      },
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123"
   }
}}
```

D.15 Creating an audio call VoiceXML message (section 6.8.5.1)

Request:

```
POST /exampleAPI/audiocall/v1/messages/voiceXml HTTP/1.1
Accept: application/json
Content-Type: application/json
Content-Length: nnnn
Host: example.com

{"voiceXMLMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "D45678",
   "clientCorrelator": "42345",
   "mediaUrl": "http://www.example.com/ann1.vxml"
}}
```

Response:

```
HTTP/1.1 201 Created
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"voiceXMLMessage": {
   "callParticipant": [
```



```
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "D45678",
   "clientCorrelator": "42345",
   "mediaUrl": "http://www.example.com/ann1.vxml",
   "messageStatusList": {
      "messageStatus": [
         {
            "callParticipant": "tel:+19585550101",
            "status": "Pending"
         },
         {
            "callParticipant": "tel:+19585550102",
            "status": "Pending"
         }
      ],
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123"
}}
```


D.16 Retrieving an active audio call VoiceXML message (section 6.9.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/messages/voiceXml/msg123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response:


```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"voiceXMLMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "D45678",
   "clientCorrelator": "42345",
   "mediaUrl": "http://www.example.com/ann1.vxml",
   "messageStatusList": {
      "messageStatus": [
         {
            "callParticipant": "tel:+19585550101",
            "status": "Played"
```



```
            },
            {
               "callParticipant": "tel:+19585550102",
               "status": "Pending"
            }
         ],
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList"
      },
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123"
}}
```


D.17 Terminating an active audio call VoiceXML message (section 6.9.6.1)

Request:

```
DELETE /exampleAPI/audiocall/v1/messages/voiceXml/msg123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response:


```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"voiceXMLMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "D45678",
   "clientCorrelator": "42345",
   "mediaUrl": "http://www.example.com/ann1.vxml",
   "messageStatusList": {
      "messageStatus": [
         {
            "callParticipant": "tel:+19585550101",
            "status": "Played"
         },
         {
            "callParticipant": "tel:+19585550102",
            "status": "Terminated"
         }
      ],
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123"
}}
```


D.18 Retrieving status of an active audio call VoiceXML message (section 6.10.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"messageStatusList": {
   "messageStatus": [
      {
         "callParticipant": "tel:+19585550101",
         "status": "Played"
      },
      {
         "callParticipant": "tel:+19585550102",
         "status": "Pending"
      }
   ],
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/voiceXml/msg123/statusList"
}}
```

D.19 Retrieving a list of all active audio call video messages (section 6.11.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/messages/video HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"messageList": {
```



```
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/video",
   "videoMessage": {
      "callParticipant": [
         "tel:+19585550101",
         "tel:+19585550102"
      ],
      "callSessionIdentifier": "E45678",
      "clientCorrelator": "42345",
      "mediaType": "video/mp4",
      "mediaUrl": "http://www.example.com/ann1.mp4",
      "messageStatusList": {
         "messageStatus": [
            {
               "callParticipant": "tel:+19585550101",
               "status": "Played"
            },
            {
               "callParticipant": "tel:+19585550102",
               "status": "Pending"
            }
         ],
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList"
      },
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/video/msg123"
   }
}}
```


D.20 Creating an audio call video message (section 6.11.5.1)

Request:


```
POST /exampleAPI/audiocall/v1/messages/video HTTP/1.1
Accept: application/json
Content-Type: application/json
Content-Length: nnnn
Host: example.com

{"videoMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "E45678",
   "clientCorrelator": "52345",
   "mediaType": "video/mp4",
   "mediaUrl": "http://www.example.com/ann1.mp4"
}}
```


Response:


```
HTTP/1.1 201 Created
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"videoMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "E45678",
   "clientCorrelator": "52345",
   "mediaType": "video/mp4",
   "mediaUrl": "http://www.example.com/ann1.mp4",
   "messageStatusList": {
      "messageStatus": [
         {
            "callParticipant": "tel:+19585550101",
            "status": "Pending"
         },
         {
            "callParticipant": "tel:+19585550102",
            "status": "Pending"
         }
      ],
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/video/msg123"
}}
```

D.21 Retrieving an active audio call video message (section 6.12.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/messages/video/msg123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"videoMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
```

```
   "callSessionIdentifier": "E45678",
   "clientCorrelator": "52345",
   "mediaType": "video/mp4",
   "mediaUrl": "http://www.example.com/ann1.mp4",
   "messageStatusList": {
      "messageStatus": [
         {
            "callParticipant": "tel:+19585550101",
            "status": "Played"
         },
         {
            "callParticipant": "tel:+19585550102",
            "status": "Pending"
         }
      ],
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/video/msg123"
}}
```


D.22 Terminating an active audio call video message (section 6.12.6.1)

Request:

```
DELETE /exampleAPI/audiocall/v1/messages/video/msg123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response:


```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"videoMessage": {
   "callParticipant": [
      "tel:+19585550101",
      "tel:+19585550102"
   ],
   "callSessionIdentifier": "E45678",
   "clientCorrelator": "52345",
   "mediaType": "video/mp4",
   "mediaUrl": "http://www.example.com/ann1.mp4",
   "messageStatusList": {
      "messageStatus": [
         {
            "callParticipant": "tel:+19585550101",
            "status": "Played"
         },
```



```
            {
               "callParticipant": "tel:+19585550102",
               "status": "Terminated"
            }
         ],
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList"
      },
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/video/msg123"
}}
```

D.23 Retrieving status of an active audio call video message (section 6.13.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/messages/video/msg123/statusList HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"messageStatusList": {
   "messageStatus": [
      {
         "callParticipant": "tel:+19585550101",
         "status": "Played"
      },
      {
         "callParticipant": "tel:+19585550102",
         "status": "Pending"
      }
   ],
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/messages/video/msg123/statusList"
}}
```

D.24 Retrieving a list of all active media capture interactions (section 6.14.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/interactions HTTP/1.1
Accept: application/json
```



```
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"interactionList": {
   "digitCapture": [
      {
         "callParticipant": "tel:+19585550101",
         "callSessionIdentifier": "F14567",
         "clientCorrelator": "62345",
         "digitConfiguration": {
            "endChar": "#",
            "maxDigits": "1",
            "minDigits": "1"
         },
         "playingConfiguration": {
            "interruptMedia": "false",
            "mediaType": "audio/mpeg",
            "messageFormat": "Audio",
            "playFileLocation": "http://www.example.com/msg1.mp3"
         },
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123"
      },
      {
         "callParticipant": "tel:+19585550104",
         "callSessionIdentifier": "F24567",
         "clientCorrelator": "72345",
         "digitConfiguration": {
            "endChar": "#",
            "maxDigits": "1",
            "minDigits": "1"
         },
         "playingConfiguration": {
            "interruptMedia": "false",
            "mediaType": "audio/mpeg",
            "messageFormat": "Audio",
            "playFileLocation": "http://www.example.com/msg3.mp3"
         },
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/collection/int456"
      }
   ],
   "recordingCapture": {
      "callParticipant": "tel:+19585550102",
      "callSessionIdentifier": "F34567",
      "clientCorrelator": "82345",
      "playingConfiguration": {
         "interruptMedia": "false",
         "mediaType": "audio/mpeg",
         "messageFormat": "Audio",
         "playFileLocation": "http://www.example.com/msg2.mp3"
      },
      "recordingConfiguration": {
```

```
            "endChar": "#",
            "maxRecordingLength": "10",
            "recFileLocation": "http://www.example.com/rec1.mp3"
         },
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/recording/int123"
      },
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions"
}}
```


D.25 Retrieving a list of all play-and-collect interactions (section 6.15.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/interactions/collection HTTP/1.1
Accept: application/json
Host: example.com
```

Response:


```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"interactionList": {
   "digitCapture": [
      {
         "callParticipant": "tel:+19585550101",
         "callSessionIdentifier": "F14567",
         "clientCorrelator": "62345",
         "digitConfiguration": {
            "endChar": "#",
            "maxDigits": "1",
            "minDigits": "1"
         },
         "playingConfiguration": {
            "interruptMedia": "false",
            "mediaType": "audio/mpeg",
            "messageFormat": "Audio",
            "playFileLocation": "http://www.example.com/msg1.mp3"
         },
         "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123"
      },
      {
         "callParticipant": "tel:+19585550104",
         "callSessionIdentifier": "F24567",
         "clientCorrelator": "72345",
         "digitConfiguration": {
            "endChar": "#",
            "maxDigits": "1",
            "minDigits": "1"
         },
```



```
          "playingConfiguration": {
             "interruptMedia": "false",
             "mediaType": "audio/mpeg",
             "messageFormat": "Audio",
             "playFileLocation": "http://www.example.com/msg3.mp3"
          },
          "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/collection/int456"
       }
   ],
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/collection"
}}
```

D.26 Playing a media file and collecting digits, returning a copy of the created resource (section 6.15.5.1)

Request:

```
POST /exampleAPI/audiocall/v1/interactions/collection HTTP/1.1
Accept: application/json
Content-Type: application/json
Content-Length: nnnn
Host: example.com

{"digitCapture": {
   "callParticipant": "tel:+19585550101",
   "callSessionIdentifier": "F14567",
   "clientCorrelator": "62345",
   "digitConfiguration": {
      "endChar": "#",
      "maxDigits": "1",
      "minDigits": "1"
   },
   "playingConfiguration": {
      "interruptMedia": "false",
      "mediaType": "audio/mpeg",
      "messageFormat": "Audio",
      "playFileLocation": "http://www.example.com/msg1.mp3"
   }
}}
```

Response:

```
HTTP/1.1 201 Created
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"digitCapture": {
   "callParticipant": "tel:+19585550101",
```

```
   "callSessionIdentifier": "F14567",
   "clientCorrelator": "62345",
   "digitConfiguration": {
      "endChar": "#",
      "maxDigits": "1",
      "minDigits": "1"
   },
   "playingConfiguration": {
      "interruptMedia": "false",
      "mediaType": "audio/mpeg",
      "messageFormat": "Audio",
      "playFileLocation": "http://www.example.com/msg1.mp3"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123"
}}
```


D.27 Playing a media file and collecting digits, returning the location of the created resource (section 6.15.5.2)

Request:

```
POST /exampleAPI/audiocall/v1/ interactions/collection HTTP/1.1
Accept: application/json
Content-Type: application/json
Content-Length: nnnn
Host: example.com

{"digitCapture": {
   "callParticipant": "tel:+19585550101",
   "callSessionIdentifier": "F14567",
   "clientCorrelator": "62345",
   "digitConfiguration": {
      "endChar": "#",
      "maxDigits": "1",
      "minDigits": "1"
   },
   "playingConfiguration": {
      "interruptMedia": "false",
      "mediaType": "audio/mpeg",
      "messageFormat": "Audio",
      "playFileLocation": "http://www.example.com/msg1.mp3"
   }
}}
```

Response:

```
HTTP/1.1 201 Created
Content-Type: application/json
```



```
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"resourceReference": {"resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123"}}
```

D.28 Retrieving an individual play-and-collect interaction (section 6.16.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/interactions/collection/int123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"digitCapture": {
   "callParticipant": "tel:+19585550101",
   "callSessionIdentifier": "F14567",
   "clientCorrelator": "62345",
   "digitConfiguration": {
      "endChar": "#",
      "maxDigits": "1",
      "minDigits": "1"
   },
   "playingConfiguration": {
      "interruptMedia": "false",
      "mediaType": "audio/mpeg",
      "messageFormat": "Audio",
      "playFileLocation": "http://www.example.com/msg1.mp3"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/collection/int123"
}}
```

D.29 Stopping interaction and removing information (section 6.16.6.1)

Request:

```
DELETE /exampleAPI/audiocall/v1/interactions/collection/int123 HTTP/1.1
Accept: application/json
```



```
Host: example.com
```

Response:

```
HTTP/1.1 204 No Content
Date: Mon, 28 Jun 2010 17:51:59 GMT
```

D.30 Retrieving a list of all play-and-record interactions (section 6.17.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/interactions/recording HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"interactionList": {
   "recordingCapture": {
      "callParticipant": "tel:+19585550102",
      "callSessionIdentifier": "F34567",
      "clientCorrelator": "82345",
      "playingConfiguration": {
         "interruptMedia": "false",
         "mediaType": "audio/mpeg",
         "messageFormat": "Audio",
         "playFileLocation": "http://www.example.com/msg2.mp3"
      },
      "recordingConfiguration": {
         "endChar": "#",
         "maxRecordingLength": "10",
         "recFileLocation": "http://www.example.com/rec2.mp3"
      },
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/recording/int123"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions"
}}
```

D.31 Playing a media file and recording response (section 6.17.5.1)

Request:

```
POST /exampleAPI/audiocall/v1/interactions/recording HTTP/1.1
Accept: application/json
Content-Type: application/json
Content-Length: nnnn
Host: example.com

{"recordingCapture": {
   "callParticipant": "tel:+19585550102",
   "callSessionIdentifier": "F34567",
   "clientCorrelator": "82345",
   "playingConfiguration": {
      "interruptMedia": "false",
      "mediaType": "audio/mpeg",
      "messageFormat": "Audio",
      "playFileLocation": "http://www.example.com/msg2.mp3"
   },
   "recordingConfiguration": {
      "endChar": "#",
      "maxRecordingLength": "10",
      "recFileLocation": "http://www.example.com/rec2.mp3"
   }
}}
```

Response:

```
HTTP/1.1 201 Created
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"recordingCapture": {
   "callParticipant": "tel:+19585550102",
   "callSessionIdentifier": "F34567",
   "clientCorrelator": "82345",
   "playingConfiguration": {
      "interruptMedia": "false",
      "mediaType": "audio/mpeg",
      "messageFormat": "Audio",
      "playFileLocation": "http://www.example.com/msg2.mp3"
   },
   "recordingConfiguration": {
       "endChar": "#",
      "maxRecordingLength": "10",
      "recFileLocation": "http://www.example.com/rec2.mp3"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/recording/int123"
}}
```

D.32 Retrieving an individual play-and-record interaction (section 6.18.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/interactions/recording/int123HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"recordingCapture": {
   "callParticipant": "tel:+19585550102",
   "callSessionIdentifier": "F34567",
   "clientCorrelator": "82345",
   "playingConfiguration": {
      "interruptMedia": "false",
      "mediaType": "audio/mpeg",
      "messageFormat": "Audio",
      "playFileLocation": "http://www.example.com/msg2.mp3"
   },
   "recordingConfiguration": {
      "endChar": "#",
      "maxRecordingLength": "10",
      "recFileLocation": "http://www.example.com/rec2.mp3"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/recording/int123"
}}
```

D.33 Stopping play-and-record interaction and removing information (section 6.18.6.1)

Request:

```
DELETE /exampleAPI/audiocall/v1/interactions/recording/int123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 204 No Content
Date: Mon, 28 Jun 2010 17:51:59 GMT
```

D.34 Retrieving a list of all recognition interactions (section 6.19.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/interactions/recognition HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"interactionList": {
   "recognitionCapture": {
      "asrConfiguration": {
         "endChar": "#",
         "scriptFileLocation": "http://www.example.com/asr.vxml",
         "scriptType": "application/voicexml+xml"
      },
      "callParticipant": "tel:+19585550102",
      "callSessionIdentifier": "G34567",
      "clientCorrelator": "92345",
      "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int123"
   },
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions"
}}
```

D.35 Engaging the participants in a dialog and automatically recognizing their spoken response; script addressed by URL (section 6.19.5.1)

Request:

```
POST /exampleAPI/audiocall/v1/interactions/recognition HTTP/1.1
Accept: application/json
Content-Type: application/json
Content-Length: nnnn
Host: example.com

{"recognitionCapture": {
   "asrConfiguration": {
      "endChar": "#",
      "scriptFileLocation": "http://www.example.com/asr.vxml",
```



```
      "scriptType": "application/voicexml+xml"
   },
   "callParticipant": "tel:+19585550102",
   "callSessionIdentifier": "G34567",
   "clientCorrelator": "92345"
}}
```


Response:

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT
```


```
{"recognitionCapture": {
   "asrConfiguration": {
      "endChar": "#",
      "scriptFileLocation": "http://www.example.com/asr.vxml",
      "scriptType": "application/voicexml+xml"
   },
   "callParticipant": "tel:+19585550102",
   "callSessionIdentifier": "G34567",
   "clientCorrelator": "92345",
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int123"
}}
```


D.36 Engaging the participants in a dialog and automatically recognizing their spoken response; script embedded (section 6.19.5.2)

Request:

```
POST /exampleAPI/audiocall/v1/interactions/recognition HTTP/1.1
Accept: application/json
Content-Type: application/json
Content-Length: nnnn
Host: example.com
```


```
{"recognitionCapture": {
   "asrConfiguration": {
      "endChar": "#",
      "script": "PHZ4bWwgeG1sbnM9Imh0dHA6Ly93d3cudzMub3JnLzIwMDEvdnhtbCI+
DQogICAgPCEtLSB2b2ljZVhhNTCBpbnN0cnVjdGlvbnMgLS0+
ICAgICAgICAgICAgDQo8L3Z4bWw+",
      "scriptType": "application/voicexml+xml",
      "scriptEncoding": "base64",
   },
   "callParticipant": "tel:+19585550102",
   "callSessionIdentifier": "H34567",
   "clientCorrelator": "a2345"
```

```
}}
```

Response:

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int123
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"resourceReference": {"resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int345"}}
```

D.37 Retrieving an individual speech recognition interaction (section 6.20.3.1)

Request:

```
GET /exampleAPI/audiocall/v1/interactions/recognition/int123HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: nnnn
Date: Mon, 28 Jun 2010 17:51:59 GMT

{"recognitionCapture": {
   "asrConfiguration": {
      "endChar": "#",
      "scriptFileLocation": "http://www.example.com/asr.vxml",
      "scriptType": "application/voicexml+xml"
   },
   "callParticipant": "tel:+19585550102",
   "callSessionIdentifier": "G34567",
   "clientCorrelator": "92345",
   "resourceURL": "http://example.com/exampleAPI/audiocall/v1/interactions/recognition/int123"
}}
```

D.38 Stopping speech recognition interaction and removing information (section 6.20.6.1)

Request:

```
DELETE /exampleAPI/audiocall/v1/interactions/recognition/int123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response:

```
HTTP/1.1 204 No Content
Date: Mon, 28 Jun 2010 17:51:59 GMT
```

Appendix E. Parlay X operations mapping (Informative)

The table below illustrates the mapping between REST resources/methods and Parlay X [3GPP 29.199-11] equivalent operations.

REST Resource	REST Method	REST Section reference	Parlay X equivalent operation
Audio Call Text Messages	POST	6.2.5	playTextMessage
Individual Audio Call Text Message	DELETE	6.3.6	endMessage
Individual Audio Call Text Message Status	GET	6.4.3	getMessageStatus
Audio Call Audio Messages	POST	6.5.5	playAudioMessage
Individual Audio Call Audio Message	DELETE	6.6.6	endMessage
Individual Audio Call Audio Message Status	GET	6.7.3	getMessageStatus
Audio Call VoiceXML Messages	POST	6.8.5	playVoiceXmlMessage
Individual Audio Call VoiceXML Message	DELETE	6.9.6	endMessage
Individual Audio Call VoiceXML Message Status	GET	6.10.3	getMessageStatus
Audio Call Video Messages	POST	6.11.5	playVideoMessage
Individual Audio Call Video Message	DELETE	6.12.6	endMessage
Individual Audio Call Video Message Status	GET	6.13.3	getMessageStatus
Play and Collect Interactions	POST	6.15.5	startPlayAndCollectInteraction
Individual Play and Collect Interaction	DELETE	6.16.6	stopMediaInteraction
Play and Record Interactions	POST	6.17.5	startPlayAndRecordInteraction
Individual Play and Record Interaction	DELETE	6.18.6	stopMediaInteraction

Table 1: Parlay X operations mapping

Appendix F. Light-weight resources (Informative)

As this version of the specification does not define any light-weight resources, this appendix is empty.

Appendix G. Authorization aspects (Normative)

This appendix specifies how to use the RESTful Audio Call API in combination with some authorization frameworks.

G.1 Use with OMA Authorization Framework for Network APIs

The RESTful Audio Call API MAY support the authorization framework defined in [Autho4API_10].

A RESTful Audio Call API supporting [Autho4API_10]:

- SHALL conform to section D.1 of [REST_NetAPI_Common];
- SHALL conform to this section G.1.

G.1.1 Scope values

G.1.1.1 Definitions

In compliance with [Autho4API_10], an authorization server serving clients requests for getting authorized access to the resources exposed by the RESTful Audio Call API:

- SHALL support the scope values defined in the table below;
- MAY support scope values not defined in this specification.

Scope value	Description	For one-time access token
oma_rest_audiocall.all_{apiVersion}	Provide access to all defined operations on the resources in this version of the API. The {apiVersion} part of this identifier SHALL have the same value as the “apiVersion” URL variable which is defined in section5.1. This scope value is the union of the other scope values listed in the next rows of this table.	No
oma_rest_audiocall.text_msg	Provide access to all defined operations regarding text messages	No
oma_rest_audiocall.audio_msg	Provide access to all defined operations regarding audio messages	No
oma_rest_audiocall.voicexml_msg	Provide access to all defined	No

Scope value	Description	For one-time access token
	operations regarding voice XML messages	
oma_rest_audiocall.video_msg	Provide access to all defined operations regarding video messages	No
oma_rest_audiocall.play_and_collect	Provide access to all defined operations regarding play-and-collect interactions	No
oma_rest_audiocall.play_and_record	Provide access to all defined operations regarding play-and-record interactions	No
oma_rest_audiocall.recognition	Provide access to all defined operations regarding speech recognition interactions	No

Table 2: Scope values for RESTful Audio Call API

G.1.1.2 Downscoping

In the case where the client requests authorization for "oma_rest_audiocall.all_{apiVersion}" scope, the authorization server and/or resource owner MAY restrict the granted scope to some of the following scope values:

- "oma_rest_audiocall.text_msg"
- "oma_rest_audiocall.audio_msg"
- "oma_rest_audiocall.voicexml_msg"
- "oma_rest_audiocall.video_msg"
- "oma_rest_audiocall.play_and_collect"
- "oma_rest_audiocall.play_and_record"
- "oma_rest_audiocall.recognition"

G.1.1.3 Mapping with resources and methods

Tables in this section specify how the scope values defined in section G.1.1.1 for the RESTful Audio Call API map to the REST resources and methods of this API. In these tables, the root "oma_rest_audiocall." of scope values is omitted for readability reasons.

Resource	URL Base URL: http://{serverRoot}/audiocall/{apiVersion}	Section refe-rence	HTTP verbs			
			GET	PUT	POST	DELETE
All audio call messages	/messages	6.1	all_{apiVersion} or text_msg or audio_msg or _voicexml_msg or video_msg (1)	n/a	n/a	n/a
Text messages	/messages/text	6.2	all_{apiVersion} or text_msg	n/a	all_{apiVersion} or text_msg	n/a
Individual text message	/messages/text/{messageId}	6.3	all_{apiVersion} or text_msg	n/a	n/a	all_{apiVersion} or text_msg
Text message status	/messages/text/{messageId}/statusList	6.4	all_{apiVersion} or text_msg	n/a	n/a	n/a
Audio messages	/messages/audio	6.5	all_{apiVersion} or audio_msg	n/a	all_{apiVersion} or audio_msg	n/a
Individual audio message	/messages/audio/{messageId}	6.6	all_{apiVersion} or audio_msg	n/a	n/a	all_{apiVersion} or audio_msg
Audio message status	/messages/audio/{messageId}/statusList	6.7	all_{apiVersion} or audio_msg	n/a	n/a	n/a
VoiceXML messages	/messages/voiceXml	6.8	all_{apiVersion} or voicexml_msg	n/a	all_{apiVersion} or voicexml_msg	n/a
Individual VoiceXML message	/messages/voiceXml/{messageId}	6.9	all_{apiVersion} or voicexml_msg	n/a	n/a	all_{apiVersion} or voicexml_msg
VoiceXML message status	/messages/voiceXml/{messageId}/statusList	6.10	all_{apiVersion} or voicexml_msg	n/a	n/a	n/a
Video messages	/messages/video	6.11	all_{apiVersion} or video_msg	n/a	all_{apiVersion} or video_msg	n/a
Individual video message	/messages/video/{messageId}	6.12	all_{apiVersion} or video_msg	n/a	n/a	all_{apiVersion} or video_msg
Video message status	/messages/video/{messageId}/statusList	6.13	all_{apiVersion} or video_msg	n/a	n/a	n/a

Table 2: Required scope values for Audio Call messages

(1) Only resources of such types that the scope allows access to will be shown when accessing the list of all audio call messages.

Resource	URL Base URL: http://{serverRoot}/audiocall/{apiVersion}	Section refe-rence	HTTP verbs			
			GET	PUT	POST	DELETE
All media capture interactions	/interactions	6.14	all_{apiVersion} or play_and_collect or play_and_record or recognition (2)	n/a	n/a	n/a
Play-and-collect interactions	/interactions/collection	6.15	all_{apiVersion} or play_and_collect	n/a	all_{apiVersion} or play_and_collect	n/a
Individual play-and-collect interaction	/interactions/collection/{interactionId}	6.16	all_{apiVersion} or play_and_collect	n/a	n/a	all_{apiVersion} or play_and_collect
Play media and record participant(s) response	/interactions/recording	6.17	all_{apiVersion} or play_and_record	n/a	all_{apiVersion} or play_and_record	n/a
Individual play-and-record interaction	/interactions/recording/{interactionId}	6.18	all_{apiVersion} or play_and_record	n/a	n/a	all_{apiVersion} or play_and_record
Engage the participant(s) in a dialog and recognize their spoken response	/interactions/recognition	6.19	all_{apiVersion} or recognition	n/a	all_{apiVersion} or recognition	n/a
Individual speech recognition interaction	/interactions/recognition/{interactionId}	6.20	all_{apiVersion} or recognition	n/a	n/a	all_{apiVersion} or recognition

Table 3: Required scope values for Audio Call interactions

(2) Only resources of such types that the scope allows access to will be shown when accessing the list of all audio call interactions.

G.1.2 Use of ‘acr:auth’

This section specifies the use of ‘acr:auth’ in place of an end user identifier (i.e. “callParticipant”) in a data structure.

An ‘acr’ URI of the form ‘acr:auth’, where ‘auth’ is a reserved keyword MAY be used to avoid exposing a real end user identifier to the application.

A client MAY use ‘acr:auth’ in a data structure in place of a user identifier (such as “callParticipant”), when the RESTful Audio Call API is used in combination with [Autho4API_10].

In the case the RESTful Audio Call API supports [Autho4API_10], the server:

- SHALL accept ‘acr:auth’ as a valid value for a user Id variable in the body (i.e. “callParticipant”).
- SHALL conform to [REST_Common_TS] section 5.8.1.1 regarding the processing of ‘acr:auth’.