Final draft ETSI ES 202 504-12 V1.1.1 (2008-02)

ETSI Standard

Open Service Access (OSA);
Parlay X Web Services;
Part 12: Multimedia Conference
(Parlay X 3)

Reference

DES/TISPAN-01034-12-OSA

Keywords

API, OSA, service

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00   Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or
perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF).
In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive
within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status.
Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Copyright Notification

ETSI

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This ETSI Standard (ES) has been produced by ETSI Technical Committee Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN), and is now submitted for the ETSI standards Membership Approval Procedure.

The present document is part 12 of a multi-part deliverable covering Open Service Access (OSA); Parlay X Web Services, as identified below:

    Part 1:   "Common";

    Part 2:   "Third Party Call";

    Part 3:   "Call Notification";

    Part 4:   "Short Messaging";

    Part 5:   "Multimedia Messaging";

    Part 6:   "Payment";

    Part 7:   "Account Management";

    Part 8:   "Terminal Status";

    Part 9:   "Terminal Location";

    Part 10:   "Call Handling";

    Part 11:   "Audio Call";

    Part 12:   "Multimedia Conference";

    Part 13:   "Address List Management";

    Part 14:   "Presence";

    Part 15:   "Message Broadcast";

    Part 16:   "Geocoding";

    Part 17:   "Application-driven Quality of Service (QoS)";

    Part 18:   "Device Capabilities and Configuration";

    Part 19:   "Multimedia Streaming Control";

    Part 20:   "Multimedia Multicast Session Management".

The present document has been defined jointly between ETSI, The Parlay Group (http://www.parlay.org) and the 3GPP.

The present document forms part of the Parlay X 3.0 set of specifications.

The present document is equivalent to 3GPP TS 29.199-12 V7.1.0 (Release 7).

1 Scope

The present document is part 12 of the Stage 3 Parlay X 3 Web Services specification for Open Service Access (OSA).

The OSA specifications define an architecture that enables application developers to make use of network functionality through an open standardized interface, i.e. the OSA APIs.

The present document specifies the Multimedia Conference Web Service. The following are defined here:

- Name spaces.

- Sequence diagrams.

- Data definitions.

- Interface specification plus detailed method descriptions.

- Fault definitions.

- Service Policies.

- WSDL Description of the interfaces.

2 References

References are either specific (identified by date of publication and/or edition number or version number) or non-specific.

- For a specific reference, subsequent revisions do not apply.

- Non-specific reference may be made only to a complete document or a part thereof and only in the following cases:

  - if it is accepted that it will be possible to use all future changes of the referenced document for the purposes of the referring document;

  - for informative references.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

For online referenced documents, information sufficient to identify and locate the source shall be provided. Preferably, the primary source of the referenced document should be cited, in order to ensure traceability. Furthermore, the reference should, as far as possible, remain valid for the expected life of the document. The reference shall include the method of access to the referenced document and the full network address, with the same punctuation and use of upper case and lower case letters.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

2.1 Normative references

The following referenced documents are indispensable for the application of the present document. For dated references, only the edition cited applies. For non-specific references, the latest edition of the referenced document (including any amendments) applies.

[1] W3C Recommendation (2 May 2001): "XML Schema Part 2: Datatypes".

NOTE: Available at http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/.

[2]   ETSI ES 202 504-1: "Open Service Access (OSA); Parlay X Web Services; Part 1: Common (Parlay X 3)".

3   Definitions and abbreviations

3.1   Definitions

For the purposes of the present document, the terms and definitions given in ES 202 504-1 [2] apply.

3.2   Abbreviations

For the purposes of the present document, the abbreviations given in ES 202 504-1 [2] apply.

4   Detailed service description

The Multimedia Conferencing is a simple Web Service that allows the creation of a multimedia conference and the dynamic management of the participants involved.

The underlying model of the service is based on the following entities:

- **Conference:** a "context" (uniquely identified) to which participants can be added/removed.

- **Participant:** each of the parties involved in the conference. There may exist a participant that is also the "owner" of the conference, i.e. the user who can end the call and/or be the reference user for billing purposes.

- **Media:** the conference can utilize multiple media streams to support the participants' communication. In particular both audio and video streams are available, including the specific stream direction (i.e. in, out, bidirectional).

NOTE: A call session allows the application to avail of other web service features that can add value to the created call session. For example the Audio Call web service can provide multimedia message delivery to call participants in the call session (playXxxMessage operation) and furthermore control of the media streams for the call participants thus enabling conversational multimedia communication including voice, video, chat, and data. Media can be added/removed for participants using the operations **addMediaForParticipants** and **deleteMediaForParticipants** in Audio Call.

An application setting up a multimedia conference must initially invoke the **createConference** operation. The result of such invocation is the creation of a "context" that represents a "virtual" room where users can "meet". A unique identifier, a callSessionIdentifier, is assigned to the just-created conference. At this stage no participant is connected.

Subsequently the application may wish to add participants to the conference. In order to do so the operation **inviteParticipant** can be used. The result of such an operation is to alert the user of the incoming connection request (e.g. the user's terminal rings).

If the application wishes to check whether the user has accepted the invitation (i.e. is connected) it can invoke (at a later time) the **getParticipantInfo** operation.

Note that:

- As soon as the first participant connects, the conference becomes "active". The duration of the conference is then measured starting from the moment the conference has became active.

- The initial media set utilized by the participant will depend on the conference type and the media actually supported by the participant's terminal.

During the conference session the application is able to:

- Add (or remove) a specific media stream to a single participant: e.g. adding a video bidirectional stream to a participant that has an audio connection to the conference. This can be obtained by invoking the media control (**addMediaForParticipants** and the **deleteMediaForParticipants)** operations of the Audio Call web service.

- Disconnect a participant from the conference, by invoking the **disconnectParticipant** operation.

- Retrieve information related to the conference and its status, by invoking **getConferenceInfo** and **getParticipants.**

There are different conditions that can determine the end of the conference:

1) The application may invoke the operation **endConference**, that "forces" the termination of the conference and the disconnection of all participants.

2) The owner of the conference (if defined) leaves the conference. If the owner is not defined this condition will apply when all the participants have left the conference (disconnected).

3) The conference duration exceeds a maximum value (specified during the conference creation step).

5   Namespaces

The Multimedia Conference interface uses the namespace:

http://www.csapi.org/wsdl/parlayx/multimedia_conference/v3_1

The data types are defined in the namespace:

http://www.csapi.org/schema/parlayx/multimedia_conference/v3_1

The "xsd" namespace is used in the present document to refer to the XML Schema data types defined in XML Schema [1]. The use of the name "xsd" is not semantically significant.

6   Sequence diagrams

The following sequence diagrams illustrate typical scenarios of interaction between an application and the Multimedia Conferencing Web Service.

6.1   Setting up a conference

Set up a multimedia conference call.

```
      : Application              : Multimedia              : User
         Logic                   Conferencing             Participants
           │                          │                       │
           │     createConference     │                       │
           ├─────────────────────────►│                       │
           │                          ├──┐                    │
           │                          │◄─┘                    │
           │                          │ Store conference info │
           │                          │                       │
           │     getConferenceInfo    │                       │
  ┌───────────────┐ ────────────────►│                       │
  │Conference status:│                │                       │
  │Initial           │                │                       │
  └───────────────┘                   │                       │
           │      inviteParticipant   │                       │
           ├─────────────────────────►│  Store participant info│
           │                          │◄─┐                    │
           │                          │  │                    │
           │                          │     Send alerting     │
           │                          ├──────────────────────►│
           │                          │                       ├──┐
           │                          │                       │◄─┘
           │                          │              Participant Ringing
           │     getParticipantInfo   │                       │
           ├─────────────────────────►│                       │
  ┌───────────────┐                    │◄─┐                    │
  │Participant status:│                │  │                    │
  │invited           │                │ Return info for participant
  └───────────────┘                   │                       │
           │     getConferenceInfo    │                       │
           ├─────────────────────────►│                       │
  ┌───────────────┐                    │◄─┐                    │
  │Conference status:│                │  │                    │
  │Initial           │                │ Return info for conference
  └───────────────┘                   │                       │
           │                          │                       ├──┐
           │                          │                       │◄─┘
           │                          │             Participant Answering
           │                          │    Participant answer │
           │                          │◄──────────────────────┤
           │                          │◄─┐                    │
           │                          │  │                    │
           │                          │ Store participant info│
           │     getParticipantInfo   │                       │
  ┌───────────────┐ ────────────────►│                       │
  │Participant status:│                │                       │
  │Connected         │                │                       │
  └───────────────┘                   │                       │
           │     getConferenceInfo    │                       │
  ┌───────────────┐ ────────────────►│                       │
  │Conference status:│                │                       │
  │Active            │                │                       │
  └───────────────┘                   │                       │
           │      inviteParticipant   │                       │
           ├─────────────────────────►│                       │
           │                          │                       │
```

Repeat previous steps to invite and connect many participants to the conference
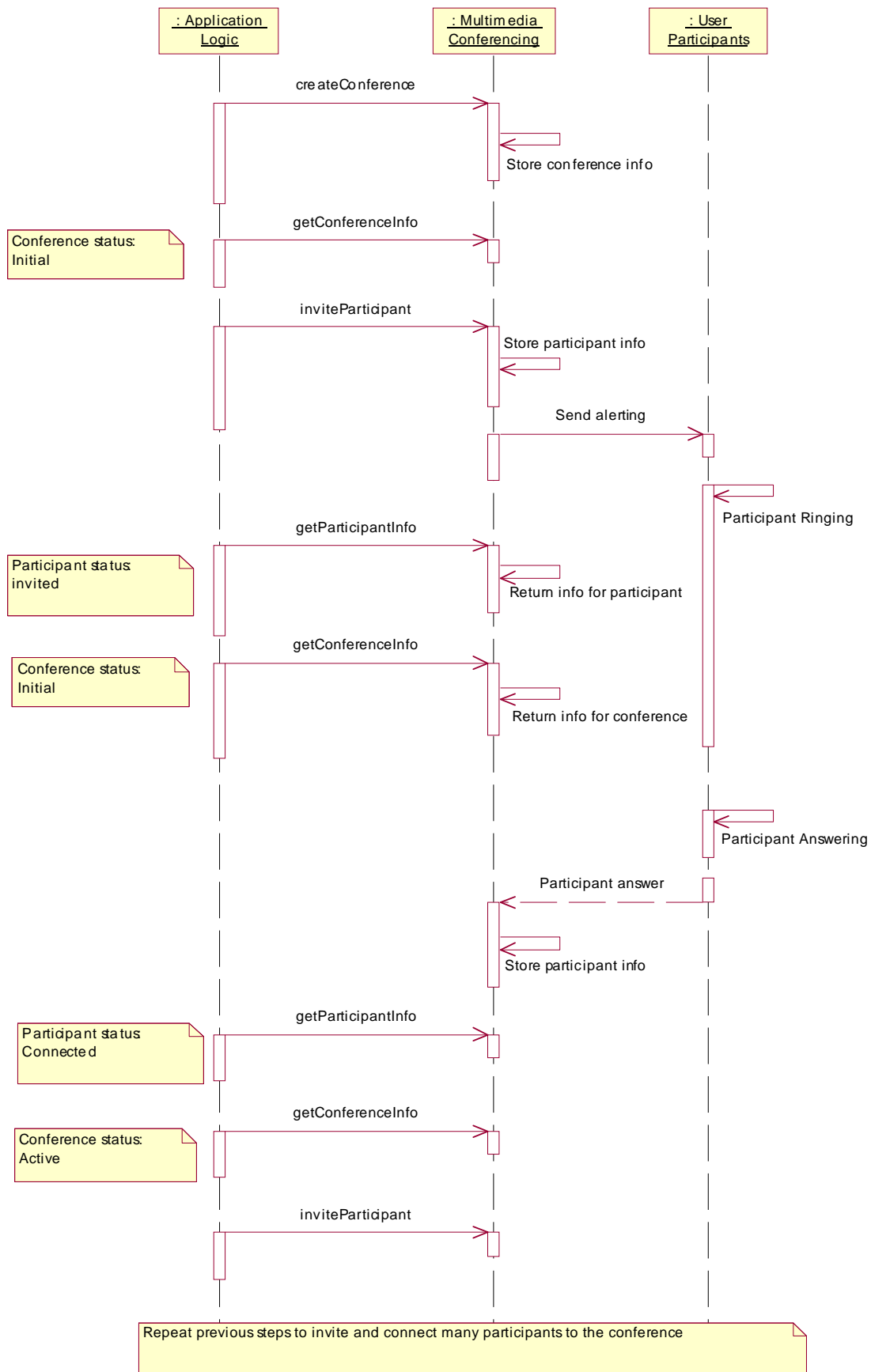
Figure 1

6.2   Void

Figure 2: Void

6.3   Conference owner disconnects

During a conference call, the conference owner disconnects.
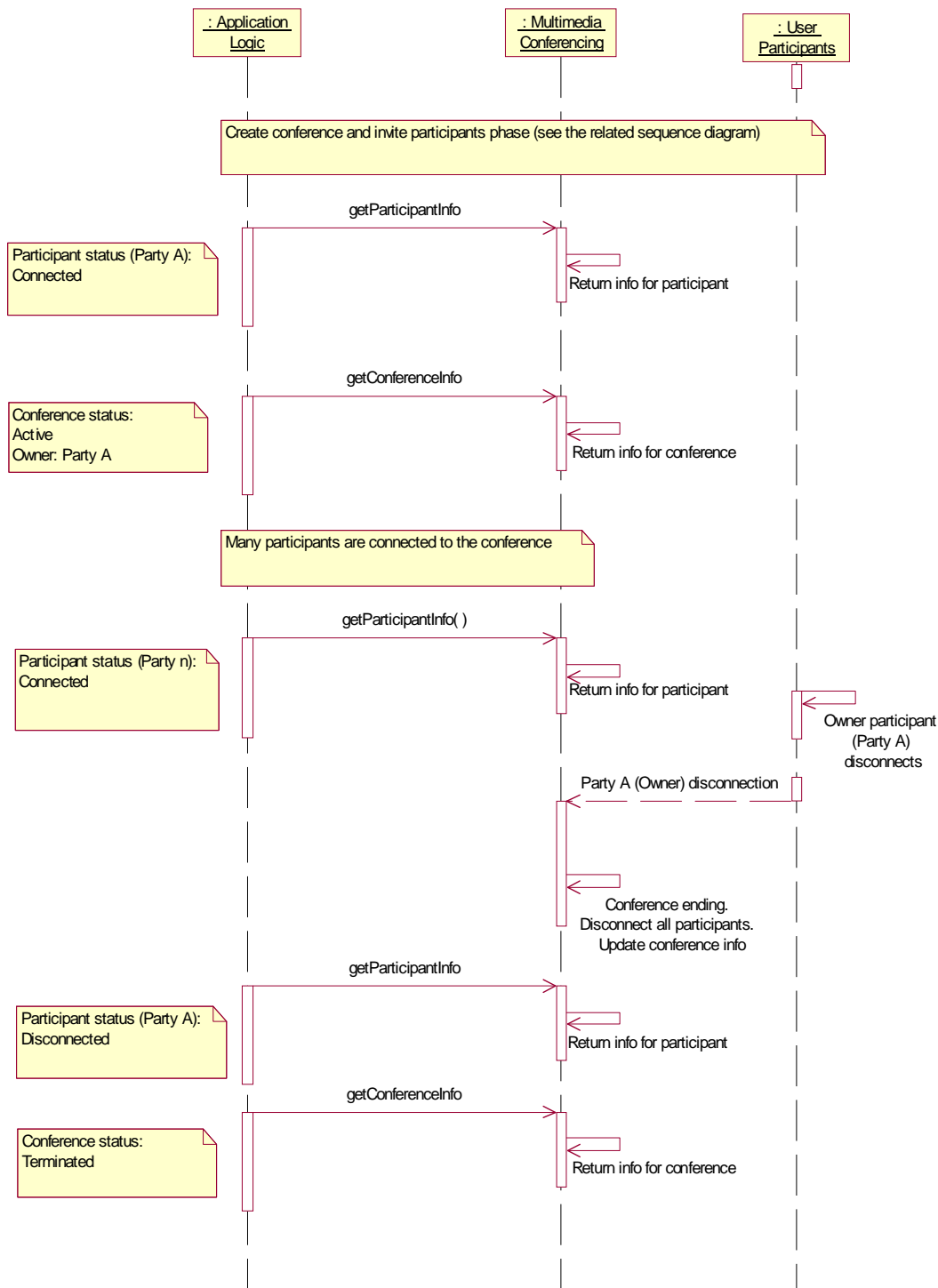

Figure 3

6.4   All participants disconnect

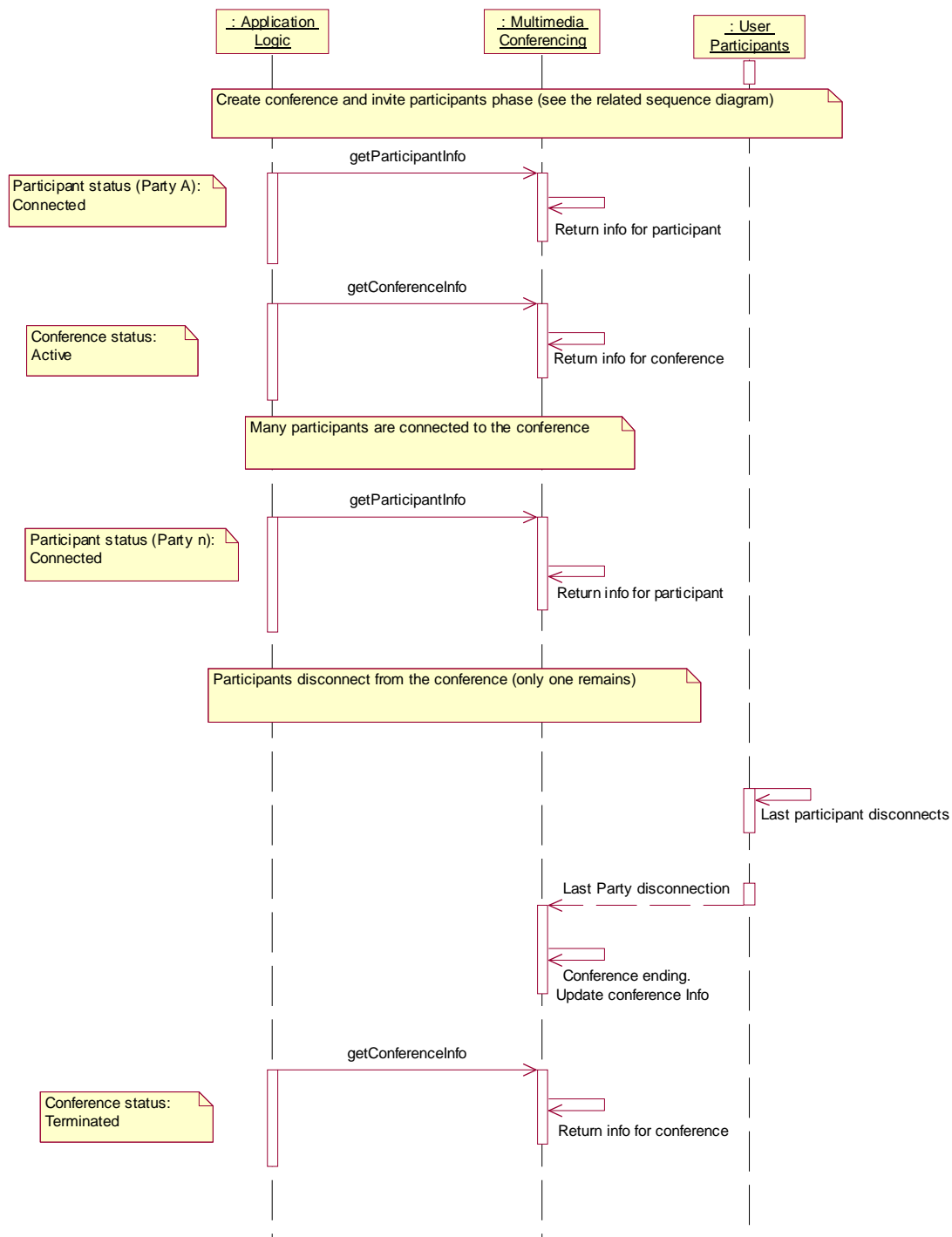End of conference call processing when all participants disconnect.

Figure 4

6.5   Conference ended by application

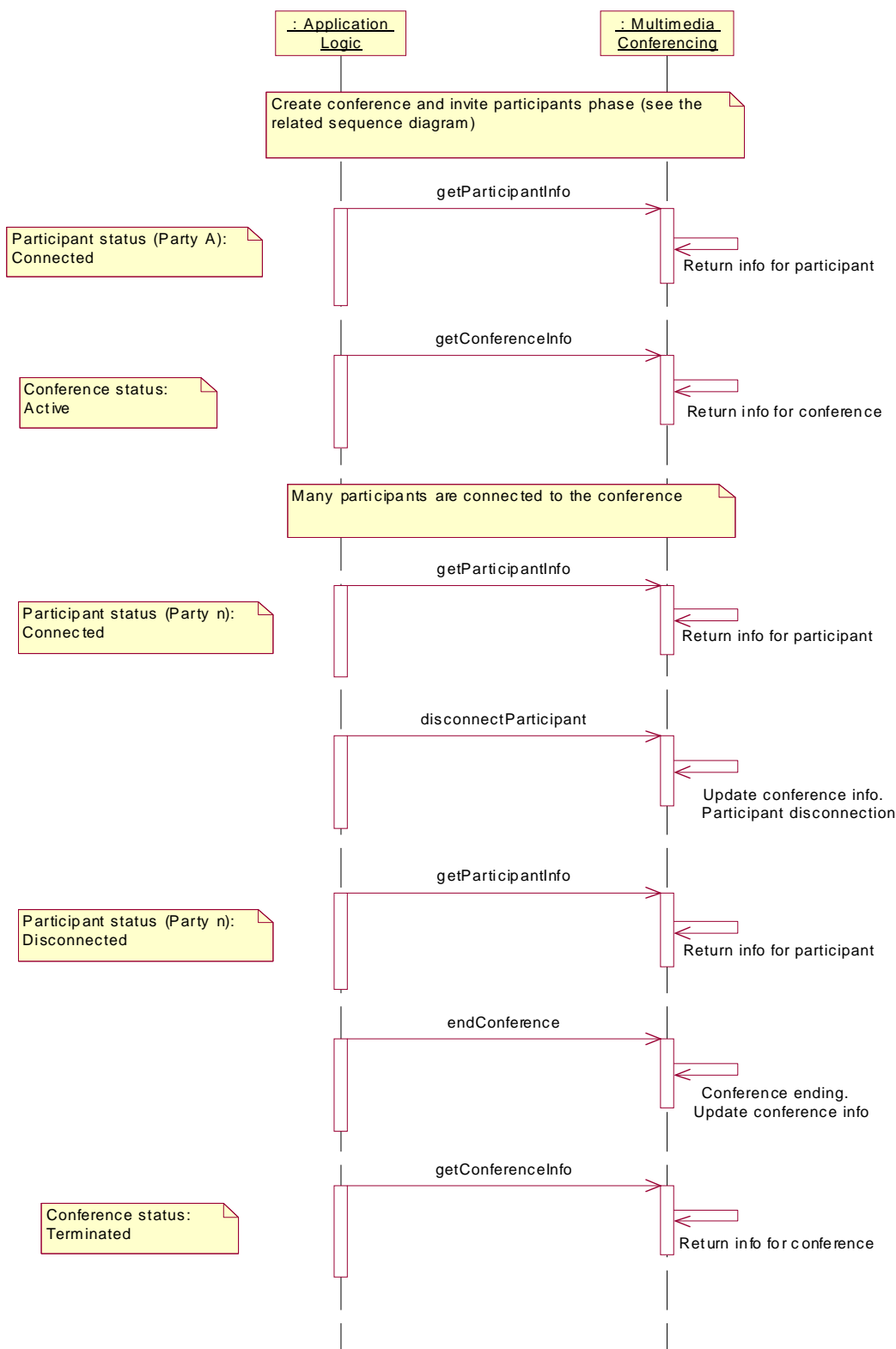End of conference call processing when the conference is ended by the application.

Figure 5

7   XML Schema data type definition

7.1   ConferenceStatus enumeration

Enumeration value	Description
Initial	The conference has been created but no participant is connected yet
Active	The conference is active, i.e. at least one user has connected
Terminated	The conference was terminated

7.2   ConferenceInfo structure

Element name	Element type	Optional	Description
status	ConferenceStatus	No	Status of the conference
startTime	xsd:dateTime	No	The time at which the conference was created
duration	xsd:int	No	The duration of the conference so far (in seconds)
owner	xsd:anyURI	No	Conference owner
numberOfParticipants	xsd:int	No	Current number of connected participants
maximumNumberOf Participants	xsd:int	No	Maximum number of participants
callSessionIdentifier	xsd:string	No	CallSessionIdentifier for the conference
conferenceDescription	xsd:string	No	Conference description

7.3   ParticipantInfo structure

Element name	Element type	Optional	Description
participant	xsd:anyURI	No	Participant identifier
mediaInfo	common:MediaInfo [1..unbounded]	No	Media information currently used
startTime	xsd:dateTime	No	Time this participant joined the conference
status	ParticipantStatus	No	Status of participant

7.4   ParticipantStatus enumeration

Enumeration value	Description
Invited	Participant invited but not connected yet
Connected	Participant connected
Disconnected	Participant disconnected

7.5   Void

7.6   Void

7.7   Void

8   Web Service interface definition

8.1   Interface: MultimediaConference

The MultimediaConference interface can be used by an application for creating a multimedia conference call and for dynamically managing the participants involved in the call.

8.1.1   Operation: createConference

The invocation of **createConference** requests to create a multi-media conference with initially no participants connected. The reference to the new multimedia conference is returned in the output parameter.

The conference termination can be driven either by a user action or by the expiring of a maximum duration. In particular, three possible situations are considered. In the first scenario, the concept of the "conference owner" is used. This user that has the control of the call and when the conference owner leaves the conference, all users are disconnected (such a user could be for instance the reference for the conference billing). In this scenario, the optional part **conferenceOwner** is present in the request message.

In the second scenario, the conference is terminated when the last participant abandons (in this case the part **conferenceOwner** is not present).

A third case is when the optional part **maximumDuration** is present: in this situation, when the maximum duration is reached, the conference is terminated.

The selection of the scenario depends on the presence of the optional parts; if no optional part is present, the conference end condition is the disconnection of the last user in conference, if both are present, the conference is terminated when the duration expires (this case could happen if the information concerning the conference owner is needed for billing purposes).

The values of the **maximumDuration** and **maximumNumberOfParticipants** parts must not exceed the corresponding service policies otherwise a policy exception is raised.

8.1.1.1   Input message: createConferenceRequest

Part name	Part type	Optional	Description
conferenceType	xsd:string	Yes	Conference type, i.e. one of a list of operator-specific identifiers that indicates how the conference is rendered on the terminals
conferenceDescription	xsd:string	No	A text describing the conference
charging	common:Charging Information	Yes	If present, defines the charge per unit of time consumed on the conference call. If the service does not support charging, a PolicyException (POL0008) will be returned
maximumDuration	xsd:int	Yes	If present it represents the maximum duration of the multimedia conference in seconds. If this part is present, it represents the end condition of the conference.
maximumNumberOf Participants	xsd:int	No	Maximum number of participants allowed
conferenceOwner	xsd:anyURI	Yes	It is the address of the multimedia conference owner. If this part is present, and the maximumDuration is not present, the conference is terminated when this user disconnects, else this information can be used for billing or other purpose

8.1.1.2   Output message: createConferenceResponse

Part name	Part type	Optional	Description
result	xsd:string	No	Call session identifier, the identifier for the created conference call

8.1.1.3   Referenced faults

ServiceException from ES 202 504-1 [2]:

- SVC0001: Service error.

- SVC0002: Invalid input value.

PolicyException from ES 202 504-1 [2]:

- POL0001: Policy error.

- POL0008: Charging not supported.

- POL0240: Too many participants.

- POL0242: Maximum duration exceeded.

8.1.2   Operation: getConferenceInfo

The invocation of **getConferenceInfo** requests the information concerning the current status of the multi-media conference call identified by **callSessionIdentifier**.

8.1.2.1   Input message: getConferenceInfoRequest

Part name	Part type	Optional	Description
callSessionIdentifier	xsd:string	No	Identifies the conference call

8.1.2.2   Output message: getConferenceInfoResponse

Part name	Part type	Optional	Description
result	ConferenceInfo	No	Status of the conference

8.1.2.3   Referenced faults

ServiceException from ES 202 504-1 [2]:

- SVC0001: Service error.

- SVC0002: Invalid input value.

PolicyException from ES 202 504-1 [2]:

- POL0001: Policy error.

8.1.3   Operation: endConference

The invocation of **endConference** requests to terminate the multi-media conference call identified by **callSessionIdentifier**.

8.1.3.1   Input message: endConferenceRequest

Part name	Part type	Optional	Description
callSessionIdentifier	xsd:string	No	Identifies the conference call

8.1.3.2 Output message: endConferenceResponse

Part name	Part type	Optional	Description
None			

8.1.3.3 Referenced faults

ServiceException from ES 202 504-1 [2]:

- SVC0001: Service error.

- SVC0002: Invalid input value.

PolicyException from ES 202 504-1 [2]:

- POL0001: Policy error.

8.1.4 Operation: inviteParticipant

The invocation of **inviteParticipant** requests to add a new participant specified by **participant** to the multi-media conference call identified by **callSessionIdentifier**. The media used for the initial connection of the new participant depends on the conference type and the participant's supported media.

The operation will fail if the conference has already reached the maximum number of participants (as specified in the creation operation).

8.1.4.1 Input message: inviteParticipantRequest

Part name	Part type	Optional	Description
callSessionIdentifier	xsd:string	No	Identifies the conference call
participant	xsd:anyURI	No	New participant invited

8.1.4.2 Output message: inviteParticipantResponse

Part name	Part type	Optional	Description
None			

8.1.4.3 Referenced faults

ServiceException from ES 202 504-1 [2]:

- SVC0001: Service error.

- SVC0002: Invalid input value.

PolicyException from ES 202 504-1 [2]:

- POL0001: Policy error.

- POL0240: Too many participants.

8.1.5 Operation: disconnectParticipant

The invocation of **disconnectParticipant** requests to disconnect the participant specified by **participant** from the multi-media conference call identified by **callSessionIdentifier**.

8.1.5.1   Input message: disconnectParticipantRequest

Part name	Part type	Optional	Description
callSessionIdentifier	xsd:string	No	Identifies the conference call
participant	xsd:anyURI	No	Participant

8.1.5.2   Output message: disconnectParticipantResponse

Part name	Part type	Optional	Description
None			

8.1.5.3   Referenced faults

ServiceException from ES 202 504-1 [2]:

- SVC0001: Service error.

- SVC0002: Invalid input value.

PolicyException from ES 202 504-1 [2]:

- POL0001: Policy error.

8.1.6   Operation: getParticipantInfo

The invocation of **getParticipantInfo** requests information concerning the current status of the participant specified by **participant**, in the multi-media conference call identified by **callSessionIdentifier**.

8.1.6.1   Input message: getParticipantInfoRequest

Part name	Part type	Optional	Description
callSessionIdentifier	xsd:string	No	Identifies the conference call
participant	xsd:anyURI	No	Participant

8.1.6.2   Output message: getParticipantInfoResponse

Part name	Part type	Optional	Description
result	ParticipantInfo	No	Status of the participant

8.1.6.3   Referenced faults

ServiceException from ES 202 504-1 [2]:

- SVC0001: Service error.

- SVC0002: Invalid input value.

PolicyException from ES 202 504-1 [2]:

- POL0001: Policy error.

8.1.7   Operation: getParticipants

The invocation of **getParticipants** requests information concerning the current status of each participant of the multi-media conference call identified by **callSessionIdentifier**. The output includes participants already disconnected from the conference (if any).

8.1.7.1   Input message: getParticipantsRequest

Part name	Part type	Optional	Description
callSessionIdentifier	xsd:string	No	Identifies the conference call

8.1.7.2   Output message: getParticipantsResponse

Part name	Part type	Optional	Description
result	ParticipantInfo [0..unbounded]	Yes	Array containing status information for each participant

8.1.7.3   Referenced faults

ServiceException from ES 202 504-1 [2]:

- SVC0001: Service error.

- SVC0002: Invalid input value.

PolicyException from ES 202 504-1 [2]:

- POL0001: Policy error.

8.1.8   Void

8.1.9   Void

9   Fault definitions

9.1   PolicyException

9.1.1   POL0240: Too many participants

Too many participants.

Name	Description
messageId	POL0240
text	Too many participants
variables	None

9.1.2   Void

PolicyException POL0241 is reserved and shall not be used.

9.1.3   POL0242: Maximum duration exceeded

Name	Description
messageId	POL0242
text	Maximum duration exceeded. Maximum allowed is %1 seconds
variables	%1 - maximum duration set by service policy

9.2   ServiceException

9.2.1   Void

ServiceException SVC0210 is reserved and shall not be used.

9.2.2   Void

ServiceException SVC0211 is reserved and shall not be used.

10   Service policies

Service policies for this service.

Name	Type	Description
MaximumDuration	common:TimeMetric	Maximum duration for which a conference may be set up.
MaximumParticipants	xsd:int	Maximum number of participants for which a conference may be set up.
ChargingSupported	xsd:boolean	Indicates whether charging is supported for the **createConference** operation

Annex A (normative):
WSDL for Multimedia Conference

The document/literal WSDL representation of this interface specification is compliant to ES 202 504-1 [2] and is contained in text files (contained in archive es_20250412v010101m0.zip) which accompany the present document.

Annex B (informative): Bibliography

ETSI TR 121 905: "Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); Vocabulary for 3GPP Specifications (3GPP TR 21.905)".

History

Document history			
V1.1.1	February 2008	Membership Approval Procedure	MV 20080425: 2008-02-26 to 2008-04-25