3GPP TSG CN Plenary Meeting #20 **NP-030248**
04-06 June 2003. Hämeenlinna, FINLAND

Source:	**CN5 (OSA)**
Title:	**Rel-6 CR 29.198-04-4 OSA API Part 4: Call control; Sub-part 4: Multi-Media Call Control SCF**
Agenda item:	**9.7**
Document for:	**APPROVAL**

Doc-1st-Level	Spec	CR	R	Ph	Subject	Cat	Ver-Curr	Doc-2nd-Level	WI
NP-030248	29.198-04-4	007	-	Rel-6	Adding QoS Reporting Functionality to MMCCS	B	5.2.0	N5-021113	OSA3
NP-030248	29.198-04-4	008	-	Rel-6	Correction of the mechanism for Requesting Event Reports in MMCCS	F	5.2.0	N5-030056	OSA3

joint-API-group (Parlay, ETSI Project OSA, 3GPP TSG_CN WG5) **N5-021113**
Meeting #21, Dublin, IRELAND, 28 – 31 October 2002

CR-Form-v7

CHANGE REQUEST

⌘ **29.198-04-4** **CR** **007** ⌘**rev** **-** ⌘ Current version: **5.2.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ □ ME □ Radio Access Network □ Core Network ☒

Title:	⌘	Adding QoS Reporting Functionality to MMCCS			
Source:	⌘	Lucent Technologies[1]			
Work item code:	⌘	OSA3	***Date:***	⌘	31/10/2002
Category:	⌘	**B**	***Release:***	⌘	*REL-6*

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Adding QoS Class reporting functionality to Multi Media Call Control API, aligned with analogous funcitonality in Data Session Control, in order to allow the application to request for notifications and reports on the QoS classes, and chnges therein, for a media stream.
Summary of change:	⌘	Adding QoS reporting functionality to Multi Media Call Control
Consequences if not approved:	⌘	

Clauses affected:	⌘	6.2, 6.4, 6.5, 6.6, 8.1.8

Other specs affected: ⌘

Y	N		
	X	Other core specifications	⌘
	X	Test specifications	
	X	O&M Specifications	

Other comments: ⌘

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

[1] Contact: Musa Unmehopa, unmehopa@lucent.com, +31 35 687 1684

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6 MultiMedia Call Control Service Interface Classes

The MultiMedia Call Control service enhances the functionality of the MultiParty Call Control Service with multi-media capabilities.

The MultiMedia Call Control Service is represented by the IpMultiMediaCallControlManager, IpMultiMediaCall, IpMultiMediaCallLeg and IpMultiMediaStream interfaces that interface to services provided by the network. Some methods are asynchronous, in that they do not lock a thread into waiting whilst a transaction performs. In this way, the client machine can handle many more calls, than one that uses synchronous message calls. To handle responses and reports, the developer must implement IpAppMultiMediaCallControlManager, IpAppMultiMediaCall and IpAppMultiMediaCallLeg to provide the callback mechanism.

To handle the multi-media aspects of a call the concept of media stream is introduced. A media stream is bi-directional media stream and is associated with a call leg. These media streams are usually negotiated between the terminals in the call. The multi-party Call Service gives the application control over the media streams associated with the legs in a multi-media call in the following way:

· the application can be triggered on the establishment of a media stream that meets the application defined characteristics.

· the application can monitor on the establishment (addition) or release (subtraction) of media streams of an ongoing call.

· the application can allow or deny the establishment of media streams (provided the stream establishment was monitored/notified in interrupt mode).

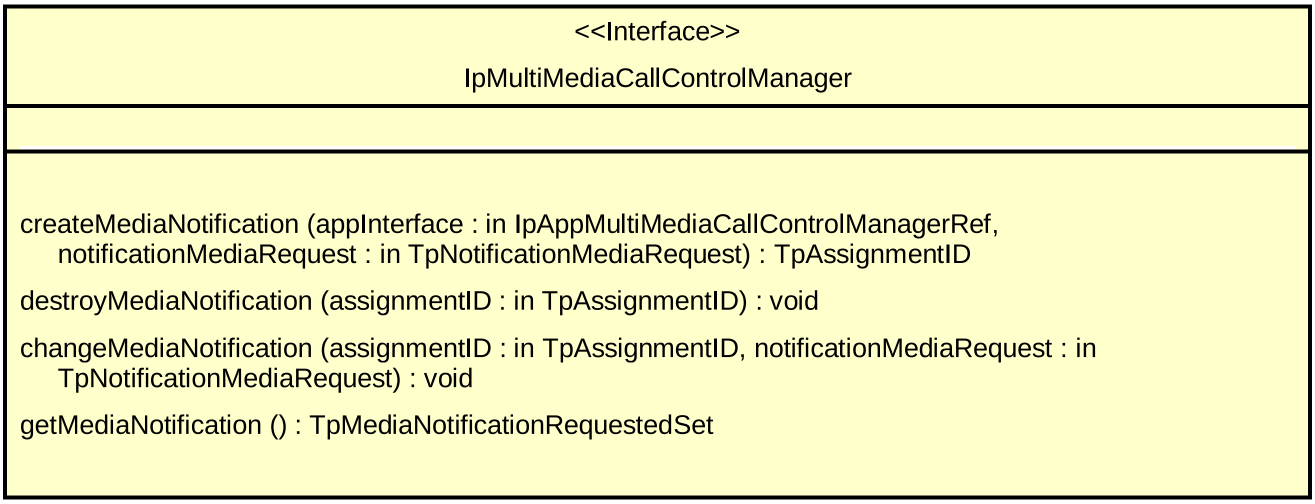
· the application can explicitly subtract already established media streams.

· the application can request the media streams associated with a specific leg.

6.1 Interface Class IpMultiMediaCallControlManager

Inherits from: IpMultiPartyCallControlManager

The Multi Media Call Control Manager is the factory interface for creating multimedia calls. The multi-media call control manager interface provides the management functions to the multi-media call control service. The application programmer can use this interface to create, destroy, change and get media stream related notifications.



<<Interface>>

IpMultiMediaCallControlManager

createMediaNotification (appInterface : in IpAppMultiMediaCallControlManagerRef, notificationMediaRequest : in TpNotificationMediaRequest) : TpAssignmentID

destroyMediaNotification (assignmentID : in TpAssignmentID) : void

changeMediaNotification (assignmentID : in TpAssignmentID, notificationMediaRequest : in TpNotificationMediaRequest) : void

getMediaNotification () : TpMediaNotificationRequestedSet

6.1.1 Method createMediaNotification()

This method is used to create media stream notifications so that events can be sent to the application.

This applies both to callsetup media (e.g., SIP initial INVITE or H.323 with faststart) and for media setup during the call.

This is the first step an application has to do to get initial notifications of media streams happening in the network. When such an event happens, the application will be informed by reportMediaNotification(). In case the application is interested in other events during the context of a particular call session it has to use the mediaStreamMonitorReq() method on the Multi-Media call leg object.

The createMediaNotification method is purely intended for applications to indicate their interest to be notified when certain media stream events take place. It is possible to subscribe to a certain media stream event for a whole range of addresses, e.g. the application can indicate it wishes to be informed when a call is made to any number starting with 800.

If some application already requested notifications with criteria that overlap the specified criteria, the request is refused with P_INVALID_CRITERIA. The criteria are said to overlap if both originating and terminating ranges overlap and the same number plan is used.

If the same application requests two notifications with exactly the same criteria but different callback references, the second callback will be treated as an additional callback. Both notifications will share the same assignmentID. The gateway will always use the most recent callback. In case this most recent callback fails the second most recent is used. In case the createMediaNotification contains no callback, at the moment the application needs to be informed the gateway will use as callback the one that has been registered by setCallback().

Returns assignmentID: Specifies the ID assigned by the multi-media call control manager interface for this newly-created notification.

Parameters

appInterface : in IpAppMultiMediaCallControlManagerRef

Specifies a reference to the application interface, which is used for callbacks.

notificationMediaRequest : in TpNotificationMediaRequest

The mediaMonitorMode is a parameter of TpMediaStreamRequest and can be in interrupt or in notify mode. If in interrupt mode the application has to specify which media streams are allowed by calling mediaStreamAllow on the callLeg.

The notificationMediaRequest parameter specifies the event specific criteria used by the application to define the event required. This is the media portion of the criteria. Only events that meet the notificationMediaRequest are reported.

Individual addresses or address ranges may be specified for the destination and/or origination.

Returns

TpAssignmentID

Raises

TpCommonExceptions, P_INVALID_CRITERIA, P_INVALID_INTERFACE_TYPE, P_INVALID_EVENT_TYPE

6.1.2 Method destroyMediaNotification()

This method is used by the application to disable Multi Media Channel notifications

Parameters

assignmentID : in TpAssignmentID

Specifies the assignment ID given by the Multi Media call control manager interface when the previous enableMediaNotification was called. If the assignment ID does not correspond to one of the valid assignment IDs, the exception P_INVALID_ASSIGNMENTID will be raised.

Raises

TpCommonExceptions

6.1.3 Method changeMediaNotification()

This method is used by the application to change the event criteria introduced with createMediaNotification. Any stored criteria associated with the specified assignmentID will be replaced with the specified criteria.

Parameters

assignmentID : in TpAssignmentID

Specifies the ID assigned by the multi-media call control manager interface for the media stream notification. If two callbacks have been registered under this assignment ID both of them will be disabled.

notificationMediaRequest : in TpNotificationMediaRequest

Specifies the new set of event specific criteria used by the application to define the event required. Only events that meet these criteria are reported.

Raises

TpCommonExceptions, P_INVALID_ASSIGNMENT_ID, P_INVALID_CRITERIA, P_INVALID_EVENT_TYPE

6.1.4 Method getMediaNotification()

This method is used by the application to query the event criteria set with createMediaNotification or changeMediaNotification.

Returns notificationsMediaRequested: Specifies the notifications that have been requested by the application.

Parameters
No Parameters were identified for this method

Returns

TpMediaNotificationRequestedSet

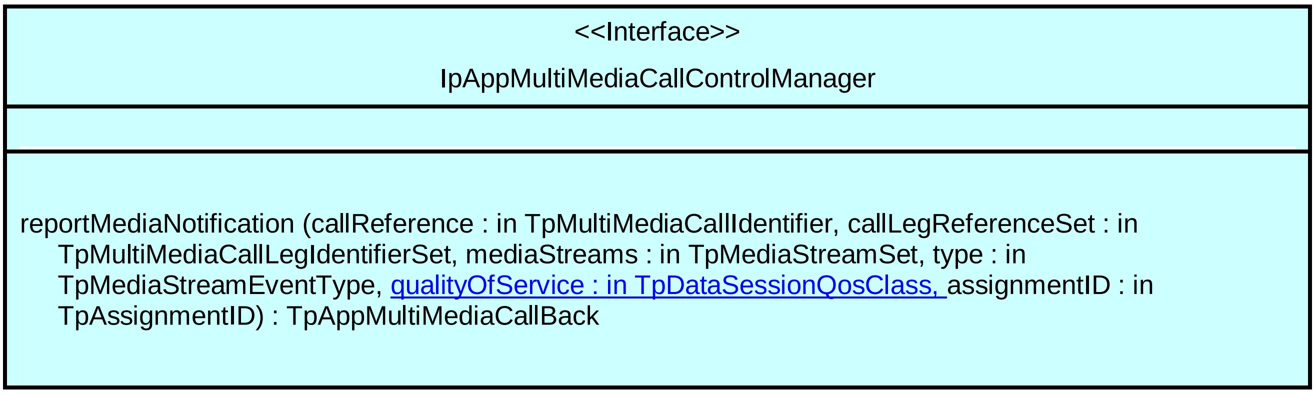
Raises

TpCommonExceptions

6.2 Interface Class IpAppMultiMediaCallControlManager

Inherits from: IpAppMultiPartyCallControlManager

The Multi Media call control manager application interface provides the application call control management functions to the multi media call control service.



<<Interface>>

IpAppMultiMediaCallControlManager

reportMediaNotification (callReference : in TpMultiMediaCallIdentifier, callLegReferenceSet : in TpMultiMediaCallLegIdentifierSet, mediaStreams : in TpMediaStreamSet, type : in TpMediaStreamEventType, qualityOfService : in TpDataSessionQosClass, assignmentID : in TpAssignmentID) : TpAppMultiMediaCallBack

6.2.1 Method reportMediaNotification()

This method is used to inform the application about the establishment of media streams.

If the corresponding monitor was in interrupt mode, then the application has to allow or deny the streams using mediaStreamAllow() method.

Returns appInterface : Specifies a reference to the application interface which implements the callback interface for the new call.

Returns appMultiMediaCallBack: Specifies references to the application interface which implements the callback interface for the new multi-media call and/or new call leg. This parameter may be null if the notification is being given in NOTIFY mode

Parameters

callReference : in TpMultiMediaCallIdentifier

Specifies the call interface on which the media streams were added or subtracted, or for which the QoS class of the media stream has changed. It also gives the corresponding sessionID.

callLegReferenceSet : in TpMultiMediaCallLegIdentifierSet

Specifies set of all callLeg references (interface and sessionID) for which the media streams were established or subtracted.

First in the set is the reference to the originating callLeg. It indicates the call leg related to the originating party. In case there is a destination call leg this will be the second leg in the set. from the notificationInfo can be found on whose behalf the notification was sent.

However, this parameter will be null if the notification is being given in NOTIFY mode

mediaStreams : in TpMediaStreamSet

Specifies all the media streams that are established. Note that this can be more media streams than requested in the createMediaNotification, e.g., when faststart is used in H.323 or in SIP when an INVITE method with SDP media stream parameters is used.

type : in TpMediaStreamEventType

Refers to the type of event on the media stream, i.e., added~~ or ~~, subtracted, or QoS class changed.

qualityOfService : in TpDataSessionQosClass

Specifies the newly negotiated Quality of Service parameters for the media stream.

assignmentID : in TpAssignmentID

Specifies the assignment id which was returned by the createMediaNotification() method. The application can use assignment id to associate events with event specific criteria and to act accordingly.

Returns

TpAppMultiMediaCallBack

6.3 Interface Class IpMultiMediaCall

Inherits from: IpMultiPartyCall

<<Interface>>

IpMultiMediaCall

superviseVolumeReq (callSessionID : in TpSessionID, volume : in TpCallSuperviseVolume, treatment : in TpCallSuperviseTreatment) : void

6.3.1 Method superviseVolumeReq()

The application calls this method to supervise a call. The application can set a granted data volume this call.

Parameters

callSessionID : in TpSessionID

Specifies the call session ID of the call.

volume : in TpCallSuperviseVolume

Specifies the granted time in milliseconds for the connection.

treatment : in TpCallSuperviseTreatment

Specifies how the network should react after the granted volume expired.

Raises

TpCommonExceptions, P_INVALID_SESSION_ID

6.4 Interface Class IpAppMultiMediaCall

Inherits from: IpAppMultiPartyCall

The application multi-media call interface contains the callbacks that will be used from the multi-media call interface for asynchronous results to requests performed by the application. The application should implement this interface.

<<Interface>>

IpAppMultiMediaCall

superviseVolumeRes (callSessionID : in TpSessionID, report : in TpCallSuperviseReport, usedVolume : in TpCallSuperviseVolume, qualityOfService : in TpDataSessionQosClass) : void

superviseVolumeErr (callSessionID : in TpSessionID, errorIndication : in TpCallError) : void

6.4.1 Method superviseVolumeRes()

This asynchronous method reports a call supervision event to the application when it has indicated its interest in these kind of events.

It is also called when the connection is terminated before the supervision event occurs. Furthermore, this method is invoked as a response to the request also when a tariff switch happens in the network during an active call or the Quality of Service parameters were renegotiated during the active call.

Parameters

callSessionID : in TpSessionID

Specifies the call session ID of the call

report : in TpCallSuperviseReport

Specifies the situation, which triggered the sending of the call supervision response.

usedVolume : in TpCallSuperviseVolume

Specifies the used time for the call supervision (in milliseconds).

qualityOfService : in TpDataSessionQosClass

Specifies the newly negotiated Quality of Service parameters for the multimedia call.

6.4.2 Method superviseVolumeErr()

This asynchronous method reports a call supervision error to the application.

Parameters

callSessionID : in TpSessionID

Specifies the call session ID of the call.

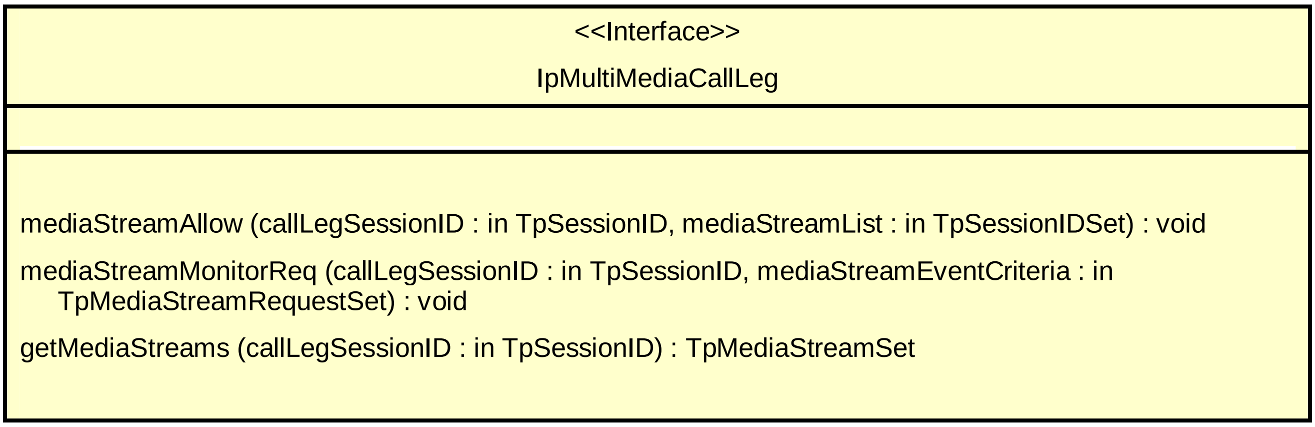
errorIndication : in TpCallError

Specifies the error which led to the original request failing.

6.5 Interface Class IpMultiMediaCallLeg

Inherits from: IpCallLeg

The Multi-Media call leg represents the signalling relationship between the call and an address. Associated with the signalling relationship there can be multiple media channels. Media channels can be started and stopped by the terminals themselves. The application can monitor on these changes and influence them.



<<Interface>>

IpMultiMediaCallLeg

mediaStreamAllow (callLegSessionID : in TpSessionID, mediaStreamList : in TpSessionIDSet) : void

mediaStreamMonitorReq (callLegSessionID : in TpSessionID, mediaStreamEventCriteria : in TpMediaStreamRequestSet) : void

getMediaStreams (callLegSessionID : in TpSessionID) : TpMediaStreamSet

6.5.1 Method mediaStreamAllow()

This method can be used to allow setup of a media stream that was reported by a mediaStreamMonitorRes method.

Parameters

callLegSessionID : in TpSessionID

Specifies the call leg session ID of the call leg.

mediaStreamList : in TpSessionIDSet

Refers to the media streams (sessionIDs) as received in the mediaStreamMonitorRes() or in the reportMediaNotification() that is allowed to be established.

Raises

TpCommonExceptions, P_INVALID_SESSION_ID

6.5.2 Method mediaStreamMonitorReq()

With this method the application can set monitors on the addition and subtraction of media streams, and a change in QoS class of media streams. The monitors can either be general or restricted to certain types of codecs.

Monitoring on addition of media streams can be done in either interrupt of notify mode. In the first case the application has to allow or deny the establishment of the stream with mediaStreamAllow.

Monitoring on subtraction of media streams is only allowed in notify mode.

Parameters

callLegSessionID : in TpSessionID

Specifies the session ID of the call leg.

mediaStreamEventCriteria : in TpMediaStreamRequestSet

Specifies the event specific criteria used by the application to define the event required. The mediaMonitorMode .is a parameter of TpMediaStreamRequest and can be in interrupt or in notify mode. If in interrupt mode the application has to respond with mediaStreamAllow().

Raises

`TpCommonExceptions, P_INVALID_SESSION_ID, P_INVALID_CRITERIA, P_INVALID_EVENT_TYPE`

6.5.3 Method getMediaStreams()

This method is used to return all currently established media streams for the leg.

Parameters

`callLegSessionID : in TpSessionID`

This method is used to return all currently open media channels for the leg,

Returns

`TpMediaStreamSet`

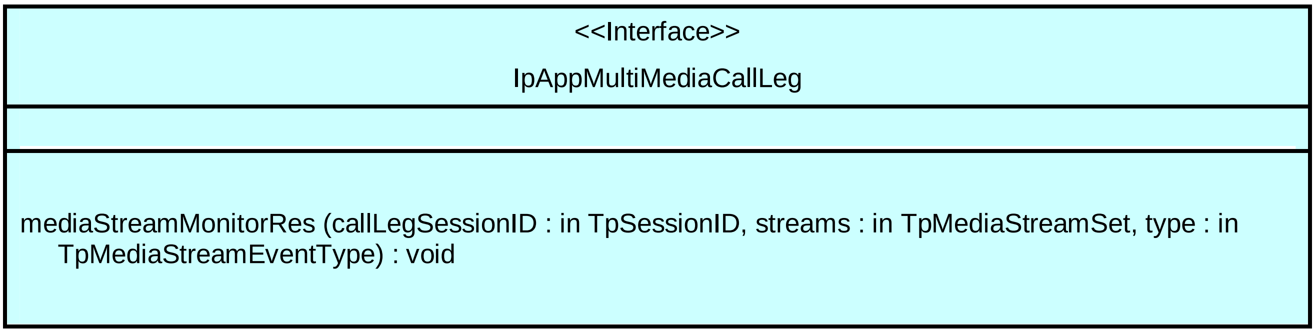
Raises

`TpCommonExceptions, P_INVALID_SESSION_ID`

6.6 Interface Class IpAppMultiMediaCallLeg

Inherits from: IpAppCallLeg

The application multi-media call leg interface contains the callbacks that will be called from the multi-media call leg for asynchronous results to requests performed by the application. The application should implement this interface.



<<Interface>>

IpAppMultiMediaCallLeg

mediaStreamMonitorRes (callLegSessionID : in TpSessionID, streams : in TpMediaStreamSet, type : in TpMediaStreamEventType) : void

6.6.1 Method mediaStreamMonitorRes()

This method is used to inform the application about the media streams that are being established (added) or subtracted, or for which the QoS class changed.

If the corresponding request was done in interrupt mode, the application has to allow or deny the media streams using mediaStreamAllow().

Parameters

callLegSessionID : in TpSessionID

Specifies the session ID of the call leg for which the media channels are opened or closed.

streams : in TpMediaStreamSet

Specifies all the media streams that are added, or for which the QoS class changed. Note that this can be more media streams than requested in the createMediaNotification, e.g., when faststart is used in H.323 or SIP INVITE with SDP media stream parameters is used.

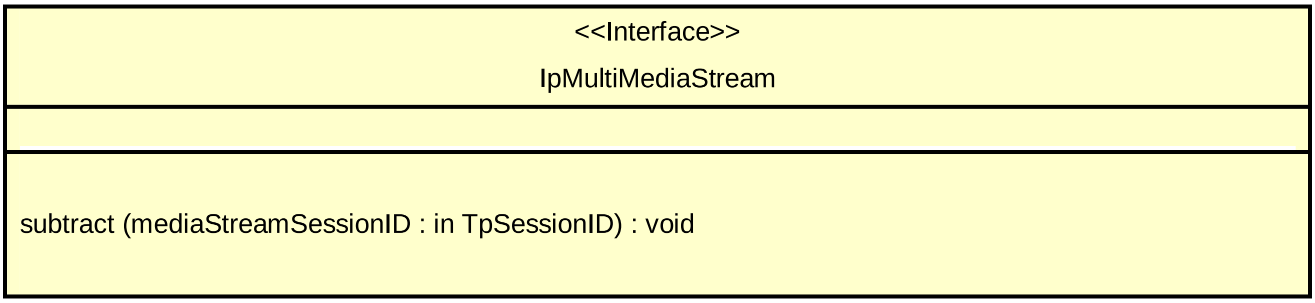
type : in TpMediaStreamEventType

Refers to the type of event on the media stream, i.e., added, ~~or~~ subtracted, or QoS class changed.

6.7 Interface Class IpMultiMediaStream

Inherits from: IpService

The Multi Media Stream Interface represents a bi-directional information stream associated with a call leg. Currently, the only available method is to subtract the media stream.



6.7.1 Method subtract()

This method can be used to subtract the multi-media stream.

Parameters

mediaStreamSessionID : in TpSessionID

Specifies the sessionID for the media stream.

Raises

TpCommonExceptions, P_INVALID_SESSION_ID

7 MultiMedia Call Control Service State Transition Diagrams

There are no State Transition Diagrams for the MultiMedia Call Control Service package

8 Multi-Media Call Control Data Definitions

This clause provides the Multi-Media call control data definitions necessary to support the API specification.

The general format of a data definition specification is described below.

- Data Type

This shows the name of the data type.

- Description

This describes the data type.

- Tabular Specification

This specifies the data types and values of the data type.

- Example

If relevant, an example is shown to illustrate the data type.

All data types referenced in the present document but not defined in this clause are defined either in the common call control data definitions in ES 202 915-4-1 or in the common data definitions which may be found in ES 202 915-2.

8.1 Event Notification Data Definitions

8.1.1 TpMediaStreamRequestSet

Defines a `Numbered Set of Data Elements` of `TpMediaStreamRequest`

8.1.2 TpMediaStreamRequest

Defines the `Sequence of Data Elements` that specify the type of media stream.

Sequence Element Name	Sequence Element Type
Direction	TpMediaStreamDirection
DataTypeRequest	TpMediaStreamDataTypeRequest
MediaMonitorMode	TpCallMonitorMode

8.1.3 TpMediaStreamDirection

Defines the direction in which the media stream is established (as seen from the leg).

Name	Value	Description
P_SEND_ONLY	0	Indicates that the offerer is only willing to send this media stream
P_RECEIVE_ONLY	1	Indicates that the offerer is only willing to receive this media stream
P_SEND_RECEIVE	2	Indicates that the offerer is willing to send and receive this media stream

8.1.4 TpMediaStreamDataTypeRequest

Defines the `Tagged Choice of Data Elements` that specify the media type and associated codecs that are of interest.

	Tag Element Type	
	TpMediaType	

Tag Element Value	Choice Element Type	Choice Element Name
P_AUDIO	TpAudioCapabilitiesType	Audio
P_VIDEO	TpVideoCapabilitiesType	Video
P_DATA	TpDataCapabilities	Data

8.1.5 TpAudioCapabilitiesType

Defines the audio codec. The requested capabilities can be indicated by adding the values together (i.e., a logical OR function). e.g., 28 indicates interest in all G.722 codes (4+8+16).

Name	Value	Description
P_G711_64K	1	g.711 on 64k, both alaw and ulaw
P_G711_56K	2	g.711 on 56k, both alaw and ulaw
P_G722_64K	4	
P_G722_56K	8	
P_G722_48K	16	
P_G7231	32	
P_G728	64	
P_G729	128	
P_G729_ANNEX_A	256	
P_IS1172	512	
P_IS1318	1024	
P_G729_ANNEXB	2048	
P_G729_ANNEX_A_AND_B	4096	
P_G7231_ANNEX_C	8192	
P_GSM_FULLRATE	16384	
P_GSM_HALFRATE	32768	
P_GSM_ENHANCED	65536	

8.1.6 TpVideoCapabilitiesType

Defines the video codec. The requested capabilities can be indicated by adding the values together (i.e., a logical OR function). e.g., 3 indicates both H.261 and H.262 codecs.

Name	Value	Description
P_H261	1	
P_H262	2	
P_H263	4	
P_IS11172	8	

8.1.7 TpDataCapabilities

A TpInt32 defining the minimum maxBitRate in bit/s. I.e., all data media streams whose maxBitRate exceeds this number are reported.

8.1.8 TpMediaStreamEventType

Defines the action performed on the media stream.

Name	Value	Description
P_MEDIA_STREAM_ADDED	0	The media stream is added
P_MEDIA_STREAM_SUBTRACTED	1	The media stream is subtracted.
P_MEDIA_STREAM_QOS_CLASS_CHANGED	2	A change in QoS class has taken place during the life of the media stream.

8.1.9 TpMediaStreamSet

Defines a Numbered Set of Data Elements of TpMediaStream

8.1.10 TpMediaStream

Defines the Sequence of Data Elements that specify the type of media stream.

Sequence Element Name	Sequence Element Type
Direction	TpMediaStreamDirection
DataType	TpMediaStreamDataType
ChannelSessionID	TpSessionID
MediaStream	IpMultiMediaStream

8.1.11 TpMediaStreamDataType

Defines the type of the reported media stream. It is identical to TpMediaStreamDataTypeRequest, only now the values are not used as a mask, but as the actual codec should be indicated for audio and video. For data the actual maximum bit rate is indicated.

8.2 Multi-Media Call Control Data Definitions

8.2.1 IpMultiMediaCall

Defines the address of an IpMultiMediaCall Interface.

8.2.2 IpMultiMediaCallRef

Defines a Reference to type IpMultiMediaCall.

8.2.3 IpAppMultiMediaCall

Defines the address of an IpAppMultiMediaCall Interface.

8.2.4 IpAppMultiMediaCallRef

Defines a Reference to type IpAppMultiMediaCall.

8.2.5 IpMultiMediaCallLeg

Defines the address of an IpMultiMediaCallLeg Interface.

8.2.6 IpMultiMediaCallLegRef

Defines a Reference to type IpMultiMediaCallLeg.

8.2.7 IpAppMultiMediaCallLeg

Defines the address of an IpAppMultiMediaCallLeg Interface.

8.2.8 IpAppMultiMediaCallLegRef

Defines a Reference to type IpAppMultiMediaCallLeg.

8.2.9 TpAppMultiMediaCallLegRefSet

Defines a Numbered Set of Data Elements of IpAppMultiMediaCallLegRef.

8.2.10 TpMultiMediaCallIdentifier

Defines the Sequence of Data Elements that unambiguously specify the MultiMediaCall object

Sequence Element Name	Sequence Element Type	Sequence Element Description
MMCallReference	IpMultiMediaCallRef	This element specifies the interface reference for the call object.
MMCallSessionID	TpSessionID	This element specifies the call session ID of the call created.

8.2.11 TpMultiMediaCallIdentifierSet

Defines a Numbered Set of Data Elements of TpMultiMediaCallIdentifier

8.2.12 TpMultiMediaCallLegIdentifier

Defines the Sequence of Data Elements that unambiguously specify the Call Leg object

Sequence Element Name	Sequence Element Type	Sequence Element Description
MMCallLegReference	IpMultiMediaCallLegRef	This element specifies the interface reference for the callLeg object.
MMCallLegSessionID	TpSessionID	This element specifies the callLeg session ID of the call created.

8.2.13 IpAppMultiMediaCallControlManager

Defines the address of an IpAppMultiMediaCallControlManager Interface.

8.2.14 IpAppMultiMediaCallControlManagerRef

Defines a Reference to type IpAppMultiMediaCallControlManager.

8.2.15 TpAppMultiMediaCallBack

Defines the Tagged Choice of Data Elements that references the application callback interfaces

	Tag Element Type	
	TpAppMultiMediaCallBackRefType	

Tag Element Value	Choice Element Type	Choice Element Name
P_APP_CALLBACK_UNDEFINED	NULL	Undefined
P_APP_MULTIMEDIA_CALL_CALLBACK	IpAppMultiMediaCallRef	AppMultiMediaCall
P_APP_CALL_LEG_CALLBACK	IpAppMultiMediaCallLegRef	AppMultiMediaCallLeg
P_APP_CALL_AND_CALL_LEG_CALLBACK	TpAppMultiMediaCallLegCallBack	AppMultiMediaCallAndCallLeg

8.2.16 TpAppMultiMediaCallBackRefType

Defines the type application call back interface.

Name	Value	Description
P_APP_CALLBACK_UNDEFINED	0	Application Call back interface undefined
P_APP_MULTIMEDIA_CALL_CALLBACK	1	Application Multi-Media Call interface referenced
P_APP_CALL_LEG_CALLBACK	2	Application Multi-Media CallLeg interface referenced
P_APP_CALL_AND_CALL_LEG_CALLBACK	3	Application Multi-Media Call and CallLeg interface referenced

8.2.17 TpAppMultiMediaCallLegCallBack

Defines the Sequence of Data Elements that references a call and a call leg application interface.

Sequence Element Name	Sequence Element Type	
AppMultiMediaCall	IpAppMultiMediaCallRef	
AppCallLegSet	TpAppMultiMediaCallLegRefSet	Specifies the set of all call leg call back references. First in the set is the reference to the call back of the originating callLeg. In case there is a call back to a destination call leg this will be second in the set.

8.2.18 TpCallSuperviseVolume

Defines the Sequence of Data Elements that specify the amount of volume that is allowed to be transmitted for the specific connection.

Sequence Element Name	Sequence Element Type	Sequence Element Description
VolumeQuantity	TpInt32	This data type is identical to a TpInt32, and defines the quantity of the granted volume that can be transmitted for the specific connection.
VolumeUnit	TpInt32	This data type is identical to a TpInt32, and defines the unit of the granted volume that can be transmitted for the specific connection. Unit must be specified as 10^n number of bytes, where n denotes the power. When the unit is for example in kilobytes, VolumeUnit must be set to 3.

8.2.19 TpNotificationMediaRequest

Defines the Sequence of Data Elements that specify the criteria for a media stream notification

Sequence Element Name	Sequence Element Type	Description
MediaNotificationScope	TpCallNotificationScope	Defines the scope of the notification request.
MediaStreamsRequested	TpMediaStreamRequestSet	Defines the media stream events which are requested

8.2.20 TpMediaNotificationRequested

Defines the Sequence of Data Elements that specify the criteria relating to event requests.

Sequence Element Name	Sequence Element Type
AppNotificationMediaRequest	TpNotificationMediaRequest
AssignmentID	TpInt32

8.2.21 TpMediaNotificationsRequestedSet

Defines a numbered Set of Data Elements of TpMediaNotificationRequested

joint-API-group (Parlay, ETSI Project OSA, 3GPP TSG_CN WG5) **N5-030056**
Meeting #22, Bangkok, THAILAND, 27 – 31 January 2003

CR-Form-v7

CHANGE REQUEST

⌘ **29.198-04-4** **CR** **008** ⌘**rev** **-** ⌘ Current version: **5.2.0** ⌘

For ***HELP*** *on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ □ ME □ Radio Access Network □ Core Network X

Title:	⌘	Correction of the mechanism for Requesting Event Reports in MMCCS		
Source:	⌘	Lucent Technologies[1]		
Work item code:	⌘	OSA3	***Date:*** ⌘	22/02/2003
Category:	⌘	**F**	***Release:*** ⌘	*REL-6*

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change: ⌘ While preparing contribution N5-021012 (Adding QoS Reporting Functionality to MMCCS) Lucent Technologies have found a potential issue with the request for notifications and event reporting. An assessment of the current method and parameter descriptions and data type definitions showed that it is currently not possible for an application to request report notifications for a specific event type. This may have been the intention, as to date there were only two event types (i.e. `P_MEDIA_STREAM_ADDED and P_MEDIA_STREAM_SUBTRACTED`). However, it could be envisaged that an application is only interested in one of the two events. Furthermore, contribution N5-021012 proposed the addition of P_MEDIA_STREAM_QOS_CLASS_CHANGED.

Summary of change: ⌘ Add an EventType field to TpMediaStreamRequest

Consequences if not approved: ⌘ There will be no way for an application to identify which event it wants to be notified of. In MPCCS this functionality is supported, because TpCallNotificationRequest in createNotification carries TpCallEventRequestSet.

Clauses affected: ⌘ 8.1.2

Other specs affected: ⌘

Y	N		
	X	Other core specifications	⌘
	X	Test specifications	
	X	O&M Specifications	

Other comments: ⌘

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

[1] Contact: Musa Unmehopa, unmehopa@lucent.com, +31 35 687 1684

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

8.1 Event Notification Data Definitions

8.1.1 TpMediaStreamRequestSet

Defines a `Numbered Set of Data Elements` of `TpMediaStreamRequest`

8.1.2 TpMediaStreamRequest

Defines the `Sequence of Data Elements` that specify the type of media stream.

Sequence Element Name	Sequence Element Type
Direction	TpMediaStreamDirection
DataTypeRequest	TpMediaStreamDataTypeRequest
MediaMonitorMode	TpCallMonitorMode
EventType	TpMediaStreamEventType

8.1.3 TpMediaStreamDirection

Defines the direction in which the media stream is established (as seen from the leg).

Name	Value	Description
P_SEND_ONLY	0	Indicates that the offerer is only willing to send this media stream
P_RECEIVE_ONLY	1	Indicates that the offerer is only willing to receive this media stream
P_SEND_RECEIVE	2	Indicates that the offerer is willing to send and receive this media stream