

**3GPP TSG CN Plenary Meeting #14
Kyoto, Japan, 12-14 December 2001**

NP-010695

Source: CN5 (OSA)
Title: Additional Rel-4 CR 29.198-04 (N5-011147)
Agenda item: 8.5
Document for: Decision

Doc-1st-Level	Spec	CR	Pha	Subject	Cat	Ver Cur	Ver -New	Doc-2nd-Level	Workitem
NP-010xyz	29.198-04	030	Rel-4	Correction of method getLastRedirectionAddress	F	4.1.0	4.2.0	N5-011147	OSA1

CHANGE REQUEST

⌘ **29.198-04 CR 030** ⌘ rev **-** ⌘ Current version: **4.1.0** ⌘

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

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title:	⌘ Correction of method getLastRedirectionAddress		
Source:	⌘ CN5		
Work item code:	⌘ OSA1	Date:	⌘ 30/112001
Category:	⌘ F	Release:	⌘ REL-4
	<i>Use <u>one</u> of the following categories:</i> 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 .		<i>Use <u>one</u> of the following releases:</i> 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)

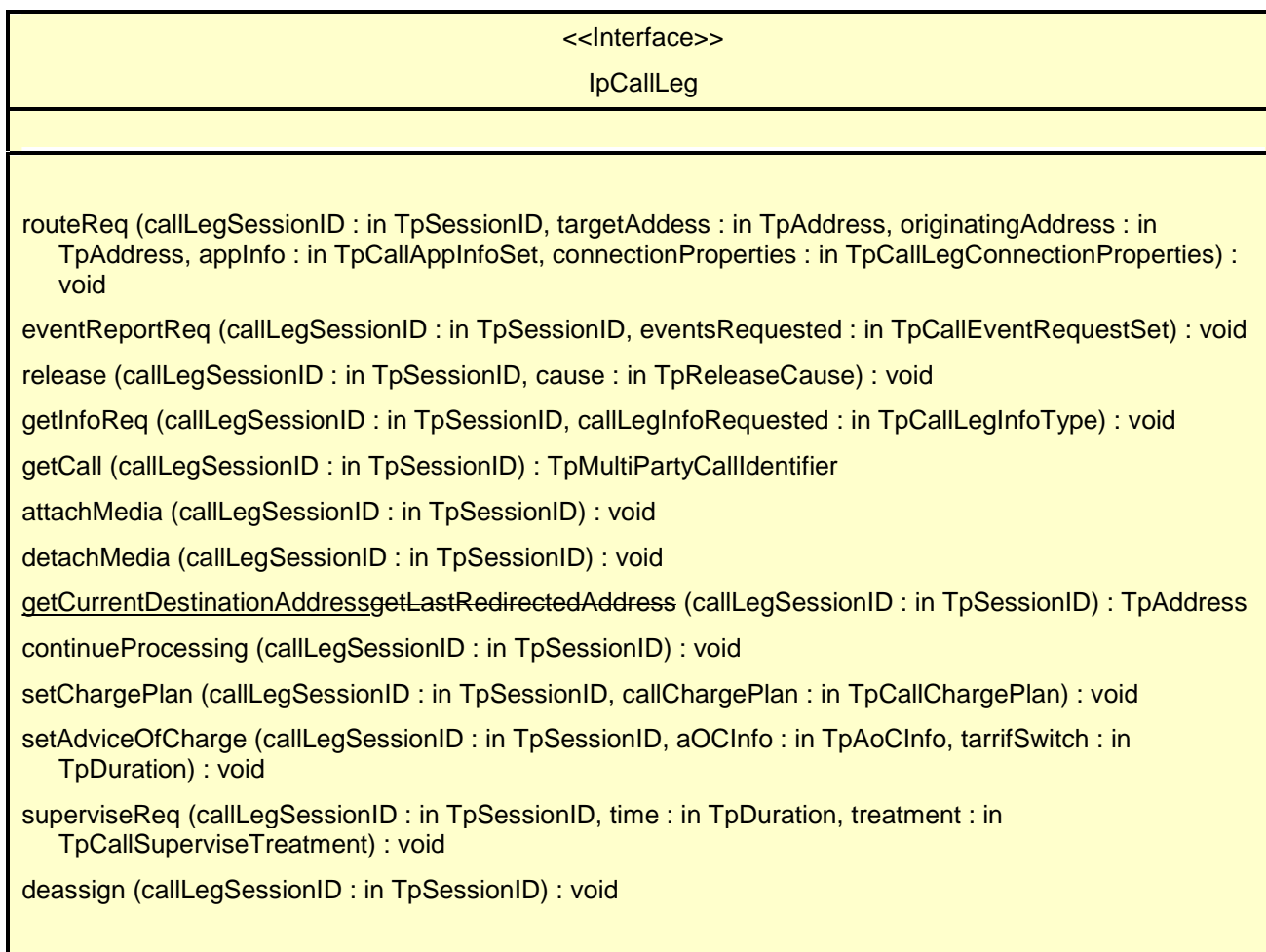
Reason for change:	⌘ Description of method getLastRedirectedAddress is ambiguous. The application should get the address of the current destination point independent if a call redirection has occurred or not in the network, i.e. the address returned to the application should not be limited to a “forwarded to” address as the description and the name of getLastRedirectedAddress method may imply. Table for allowed methods for Originating Call Leg STD includes getLastRedirectedAddress. The method description tells it is not allowed to be invoked on the Originating Call Leg STD.
Summary of change:	⌘ Name of getLastRedictedAddress changed to getCurrentDestinationAddress and method description corrected to reflect that reported address is the current destination address. In the table for allowed methods for Originating Call Leg STD getLastRedirectedAddress is deleted. In the table for allowed methods for Terminating Call Leg STD getLastRedirectedAddress is changed to getCurrentDestinationAddress.
Consequences if not approved:	⌘ Ambiguous behaviour for getLastRedirectedAddress. Not correcting this now could lead to future backwards compatibilbity problems with early implementations.

Clauses affected:	⌘ 6.9.5, 6.10.3, 6.10.3.1.5, 6.10.3.2.4		
Other specs affected:	⌘ <input type="checkbox"/> Other core specifications <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications	⌘	
Other comments:	⌘		

6.9.5 Interface Class IpCallLeg

Inherits from: IpService

The call leg interface represents the logical call leg associating a call with an address. The call leg tracks its own states and allows charging summaries to be accessed. The leg represents the signalling relationship between the call and an address. An application that uses the IpCallLeg interface to set up connections has good control, e.g. by defining leg specific event request and can obtain call leg specific report and events.



Method

routeReq ()

This asynchronous method requests routing of the call leg to the remote party indicated by the targetAddress.

In case the connection to the destination party is established successfully the CallLeg will be either detached or attached to the call based on the attach Mechanism values specified in the connectionProperties parameter.

The extra address information such as originatingAddress is optional. If not present (i.e. the plan is set to P_ADDRESS_PLAN_NOT_PRESENT), the information provided in the corresponding addresses from the route is used, otherwise network or gateway provided addresses will be used.

If the application wishes that the call leg should be represented in the network as being a redirection it should include a value for the field P_CALL_APP_ORIGINAL_DESTINATION_ADDRESS of TpCallAppInfo.

This operation continues processing of the call leg.

*Parameters***callLegSessionID : in TpSessionID**

Specifies the call leg session ID of the call leg.

targetAddress : in TpAddress

Specifies the destination party to which the call leg should be routed

originatingAddress : in TpAddress

Specifies the address of the originating (calling) party.

appInfo : in TpCallAppInfoSet

Specifies application-related information pertinent to the call leg (such as alerting method, tele-service type, service identities and interaction indicators).

connectionProperties : in TpCallLegConnectionProperties

Specifies the properties of the connection.

*Raises***TpCommonExceptions, P_INVALID_SESSION_ID, P_INVALID_NETWORK_STATE,
P_INVALID_ADDRESS, P_UNSUPPORTED_ADDRESS_PLAN***Method***eventReportReq()**

This asynchronous method sets, clears or changes the criteria for the events that the call leg object will be set to observe.

*Parameters***callLegSessionID : in TpSessionID**

Specifies the call leg session ID of the call leg.

eventsRequested : in TpCallEventRequestSet

Specifies the event specific criteria used by the application to define the events required. Only events that meet these criteria are reported. Examples of events are "address analysed", "answer", "release".

*Raises***TpCommonExceptions, P_INVALID_SESSION_ID, P_INVALID_EVENT_TYPE,
P_INVALID_CRITERIA***Method***release()**

This method requests the release of the call leg. If successful, the associated address (party) will be released from the call, and the call leg deleted. Note that in some cases releasing the party may lead to release of the complete call in the network. The application will be informed of this with callEnded().

This operation continues processing of the call leg.

*Parameters***callLegSessionID : in TpSessionID**

Specifies the call leg session ID of the call leg.

cause : in TpReleaseCause

Specifies the cause of the release.

*Raises***TpCommonExceptions, P_INVALID_SESSION_ID, P_INVALID_NETWORK_STATE***Method***getInfoReq()**

This asynchronous method requests information associated with the call leg to be provided at the appropriate time (for example, to calculate charging). Note: in the call leg information must be accessible before the objects of concern are deleted.

*Parameters***callLegSessionID : in TpSessionID**

Specifies the call leg session ID of the call leg.

callLegInfoRequested : in TpCallLegInfoType

Specifies the call leg information that is requested.

*Raises***TpCommonExceptions, P_INVALID_SESSION_ID***Method***getCall()**

This method requests the call associated with this call leg.

Returns callReference: Specifies the interface and sessionID of the call associated with this call leg.

*Parameters***callLegSessionID : in TpSessionID**

Specifies the call leg session ID of the call leg.

*Returns***TpMultiPartyCallIdentifier***Raises***TpCommonExceptions, P_INVALID_SESSION_ID***Method***attachMedia()**

This method requests that the call leg be attached to its call object. This will allow transmission on all associated bearer connections or media streams to and from other parties in the call. The call leg must be in the connected state for this method to complete successfully.

Parameters

callLegSessionID : in TpSessionID

Specifies the sessionID of the call leg to attach to the call.

Raises

TpCommonExceptions, P_INVALID_SESSION_ID, P_INVALID_NETWORK_STATE

Method

detachMedia()

This method will detach the call leg from its call, i.e., this will prevent transmission on any associated bearer connections or media streams to and from other parties in the call. The call leg must be in the connected state for this method to complete successfully.

Parameters

callLegSessionID : in TpSessionID

Specifies the sessionID of the call leg to detach from the call.

Raises

TpCommonExceptions, P_INVALID_SESSION_ID, P_INVALID_NETWORK_STATE

Method

getCurrentDestinationAddressgetLastRedirectedAddress()

Queries the current address of the destination ~~last address~~ the leg has been redirected to.

Returns the address of the destination point towards which the call leg has been routed. ~~redirectedAddress: Specifies the last address where the call leg was redirected to.~~

If this method is invoked on the Originating Call Leg, exception P_INVALID_STATE will be thrown.

Parameters

callLegSessionID : in TpSessionID

Specifies the call session ID of the call leg.

Returns

TpAddress

Raises

TpCommonExceptions, P_INVALID_SESSION_ID

6.10.3 State Transition Diagrams for IpCallLeg

6.10.3.1.5 Overview of allowed methods, Originating Call Leg STD

state	methods allowed
Initiating	attachMedia (as a request), detachMedia, (as a request) getCall , getLastRedirectedAddress, continueProcessing, release (call leg), deassign eventReportReq, getInfoReq, setChargePlan, setAdviceOfCharge, superviseReq
Analysing	attachMedia (as a request), detachMedia, (as a request) getCall , getLastRedirectedAddress, continueProcessing, release (call leg), deassign eventReportReq, getInfoReq, setChargePlan, setAdviceOfCharge, superviseReq
Active	attachMedia, detachMedia, getCall , getLastRedirectedAddress, continueProcessing, release deassign eventReportReq, getInfoReq, setChargePlan, setAdviceOfCharge, superviseReq
Releasing	getCall , getLastRedirectedAddress, continueProcessing, release deassign

6.10.3.2.4 Overview of allowed methods and trigger events, Terminating Call Leg STD

state	methods allowed
Idle	routeReq, getCall , <u>getCurrentDestinationAddress</u> <u>getLastRedirectedAddress</u> , release, deassign eventReportReq, getInfoReq, setChargePlan, setAdviceOfCharge, superviseReq
Active	attachMedia detachMedia getCall , <u>getCurrentDestinationAddress</u> <u>getLastRedirectedAddress</u> , continueProcessing, release, deassign eventReportReq, getInfoReq, setChargePlan, setAdviceOfCharge, superviseReq
Releasing	- getCall , <u>getCurrentDestinationAddress</u> <u>getLastRedirectedAddress</u> , continueProcessing, release, deassign