3GPP TSG CN Plenary Meeting #17 **NP-020424**
4 - 6 September 2002, Biarritz, FRANCE

Source: **CN5 (OSA)**

Title: **Rel-4 CRs 29.198-04 OSA API Part 4: Call control**

Agenda item: **7.10**

Document for: **APPROVAL**

Doc-1st-Level	Spec	CR	Rev	Phase	Subject	Cat	Version-Current	Doc-2nd-Level	Workitem
NP-020424	29.198-04	057	-	Rel-4	Correction on use of NULL in Call Control API	F	4.4.0	N5-020762	OSA1

joint API group (Parlay, ETSI Project OSA, 3GPP TSG_CN WG5) N5-020762
Meeting #19, Montreal, CANADA, 8 – 12 July 2002

CR-Form-v5

CHANGE REQUEST

⌘	**29.198-04**	**CR**	**057**	⌘ **rev**	**-**	⌘	Current version:	**4.4.0**	⌘

*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 on use of NULL in Call Control API		
Source:	⌘	CN5		
Work item code:	⌘	OSA1	***Date:*** ⌘	12/07/2002
Category:	⌘	**F**	***Release:*** ⌘	*REL-4*

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)*

Reason for change:	⌘	OMG IDL does not support NULL as a valid value for a data type; attempts to send a null value result in a marshalling exception and a gateway can never receive the call.
Summary of change:	⌘	Occurrences of the use of NULL as a valid setting for Call Control API parameters have been replaced. Use of null for structs TpCall*Identifier modified to define appropriate behaviour in NOTIFY mode.
Consequences if not approved:	⌘	Failure to correct the API shall result in vendor specific interpretation and interoperability issues.

Clauses affected:	⌘	6.3.2; 7.3.2;	
Other specs affected:	⌘	□ Other core specifications □ Test specifications □ O&M Specifications	⌘
Other comments:	⌘	Mirror in Rel-5 CR 29.198-04-2 (N5-020765)	

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.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.

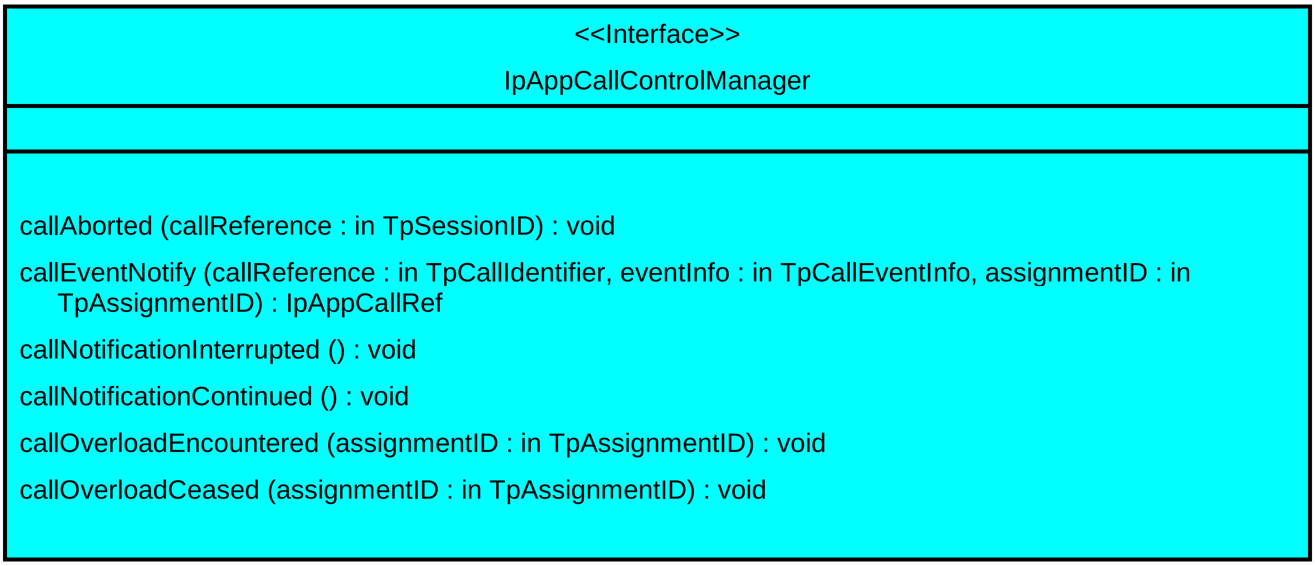
3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

*********************************** START OF FIRST CHANGE *************************************

6.3.2 Interface Class IpAppCallControlManager

Inherits from: IpInterface

The generic call control manager application interface provides the application call control management functions to the generic call control service.



<<Interface>>

IpAppCallControlManager

callAborted (callReference : in TpSessionID) : void

callEventNotify (callReference : in TpCallIdentifier, eventInfo : in TpCallEventInfo, assignmentID : in TpAssignmentID) : IpAppCallRef

callNotificationInterrupted () : void

callNotificationContinued () : void

callOverloadEncountered (assignmentID : in TpAssignmentID) : void

callOverloadCeased (assignmentID : in TpAssignmentID) : void

Method
callAborted()

This method indicates to the application that the call object (at the gateway) has aborted or terminated abnormally. No further communication will be possible between the call and application.

Parameters

callReference : in TpSessionID

Specifies the sessionID of call that has aborted or terminated abnormally.

Method
callEventNotify()

This method notifies the application of the arrival of a call-related event.

If this method is invoked with a monitor mode of P_CALL_MONITOR_MODE_INTERRUPT, then the APL has control of the call. If the APL does nothing with the call (including its associated legs) within a specified time period (the duration of which forms a part of the service level agreement), then the call in the network shall be released and callEnded() shall be invoked, giving a release cause of 102 (Recovery on timer expiry).

When this method is invoked with a monitor mode of P_CALL_MONITOR_MODE_INTERRUPT, the application writer should ensure that no routeReq() is performed until an IpAppCall has been passed to the gateway, either through an explicit setCallback() invocation on the supplied IpCall, or via the return of the callEventNotify() method.

Returns appCall: Specifies a reference to the application interface which implements the callback interface for the new call. ~~This parameterwill be null if the notification is in NOTIFY mode.~~ If the application has previously explicitly passed a reference to the IpAppCall interface using a setCallback() invocation, this parameter may be null, or if supplied must be the same as that provided during the setCallback().

This parameter will be null if the notification is in NOTIFY mode.

Parameters

callReference : in TpCallIdentifier

Specifies the reference to the call interface to which the notification relates. ~~This parameter will be null~~ I~~i~~f the notification is in NOTIFY mode, this parameter shall be ignored by the application client implementation, and consequently the implementation of the SCS entity invoking callEventNotify may populate this parameter as it chooses.

eventInfo : in TpCallEventInfo

Specifies data associated with this event.

assignmentID : in TpAssignmentID

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

Returns

IpAppCallRef

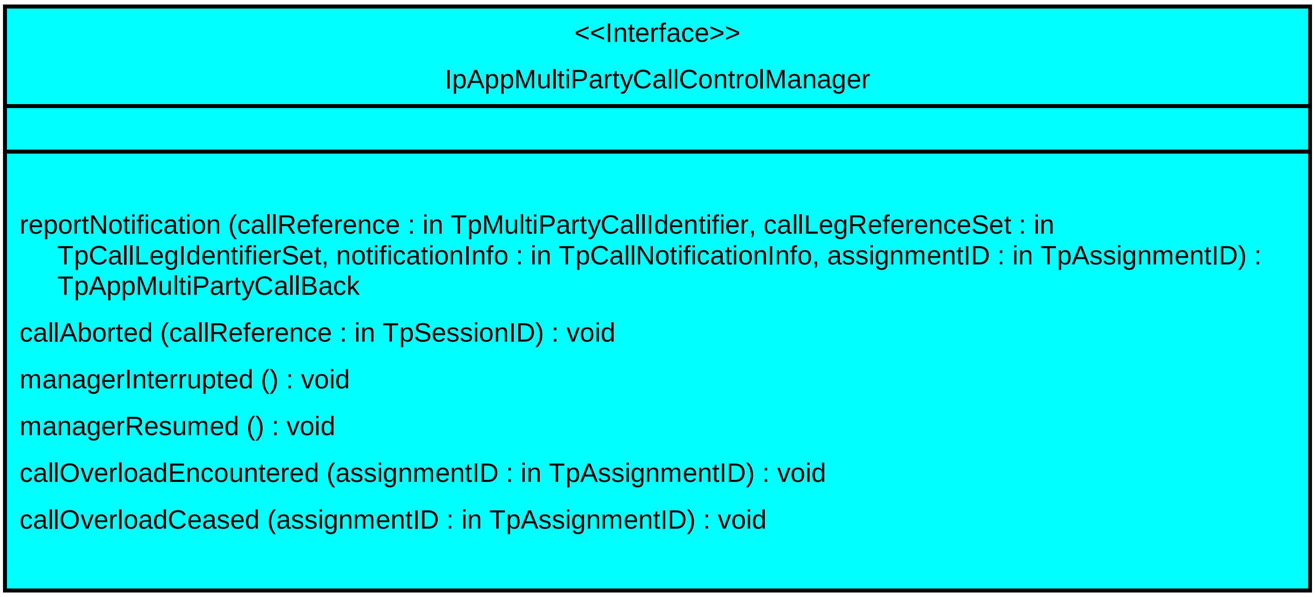
********************************* END OF FIRST CHANGE *********************************

********************************* START OF SECOND CHANGE *********************************

7.3.2 Interface Class IpAppMultiPartyCallControlManager

Inherits from: IpInterface

The Multi-Party call control manager application interface provides the application call control management functions to the Multi-Party call control service.



<<Interface>>

IpAppMultiPartyCallControlManager

reportNotification (callReference : in TpMultiPartyCallIdentifier, callLegReferenceSet : in TpCallLegIdentifierSet, notificationInfo : in TpCallNotificationInfo, assignmentID : in TpAssignmentID) : TpAppMultiPartyCallBack

callAborted (callReference : in TpSessionID) : void

managerInterrupted () : void

managerResumed () : void

callOverloadEncountered (assignmentID : in TpAssignmentID) : void

callOverloadCeased (assignmentID : in TpAssignmentID) : void

Method

reportNotification()

This method notifies the application of the arrival of a call-related event.

If this method is invoked with a monitor mode of P_CALL_MONITOR_MODE_INTERRUPT, then the APL has control of the call. If the APL does nothing with the call (including its associated legs) within a specified time period (the duration of which forms a part of the service level agreement), then the call in the network shall be released and callEnded() shall be invoked, giving a release cause of P_TIMER_EXPIRY.

Returns appCallBack: Specifies references to the application interface which implements the callback interface for the new call and/or new call leg. ~~This parameter may be null if the notification is being given in NOTIFY mode.~~ If the application has previously explicitly passed a reference to the callback interface using a setCallback() invocation, this parameter may be set to P_APP_CALLBACK_UNDEFINED, or if supplied must be the same as that provided during the setCallback().

This parameter will be set to P_APP_CALLBACK_UNDEFINED if the notification is in NOTIFY mode.

Parameters

callReference : in TpMultiPartyCallIdentifier

Specifies the reference to the call interface to which the notification relates. ~~This parameter will be null i~~If the notification is being given in NOTIFY mode, this parameter shall be ignored by the application client implementation, and consequently the implementation of the SCS entity invoking reportNotification may populate this parameter as it chooses.~~.~~

callLegReferenceSet : in TpCallLegIdentifierSet

Specifies the set of all call leg references. 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, this parameter shall be ignored by the application client implementation, and consequently the implementation of the SCS entity invoking reportNotification may populate this parameter as it chooses.

notificationInfo : in TpCallNotificationInfo

Specifies data associated with this event (e.g. the originating or terminating leg which reports the notification).

`assignmentID : in TpAssignmentID`

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

Returns

`TpAppMultiPartyCallBack`

********************************** END OF SECOND CHANGE ************************************