

CHANGE REQUEST

⌘ **29.198-04-3 CR 012** ⌘ rev **-1** ⌘ 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: ⌘ (U)SIM ME/UE Radio Access Network Core Network

| | | | |
|--|--|---|--------------|
| Title: | ⌘ Unclear overlap criteria for rejection of createNotification | | |
| Source: | ⌘ Appium | | |
| Work item code: | ⌘ OSA3 | Date: | ⌘ 06/11/2002 |
| Category: | ⌘ F C | Release: | ⌘ REL-6 |
| Use <u>one</u> 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 <u>one</u> of the following releases: 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REI-4 (Release 4) REI-5 (Release 5) Rel-6 (Release 6) | |

| | |
|--------------------------------------|--|
| Reason for change: | ⌘ Unclarity about when a createNotification for MPCC can be refused. The OSA Specification currently does not consider the possibility that a network may allow more applications to be invoked on the same call or session. The current restriction in the OSA specification originate from the Single Point of Control principle as defined in IN CS1 allowing only one service at a time to control a call. |
| Summary of change: | ⌘ Add of clarity to the current notification overlap criteria handling describing when a rejection of createNotification may occur. This also caters for the option that the core network may support multi services, i.e. a Multiple Points of Control concept enabling multiple applications to act on the same call or session. The change proposed is solely in the text part. |
| Consequences if not approved: | ⌘ Not clear when a createNotification for MPCC can be refused with P_INVALID_CRITERIA to the client application. This may cause undesired restrictions in the allowed number of applications that can be invoked on a call or session - and hereby cause unnecessary rejection of applications that could have been allowed. It prevents more opportunities for traffic in the networks. |

| | |
|------------------------------|--|
| Clauses affected: | ⌘ 6.1 |
| Other specs affected: | ⌘ <input checked="" type="checkbox"/> Other core specifications ⌘ 29.198-04-3 <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications |
| Other comments: | ⌘ Correlated with 29.198-04-3 CR in N5-021159 |

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.

Introduction

It is not clear when a createNotification for MPCC can be refused with P_INVALID_CRITERIA to the client application . There have in the past been raised some concern about the criteria overlap definition and the associated restriction not to allow more than one application to control a call or session.

Add text to clarify when createNotification will be rejected due to overlapping criteria and the option to overrule this criteria check to enable muti services support.

Part 4-3

Changes to 3GPP TS 29.198-04-3

6 MultiParty Call Control Service Interface Classes

The Multi-party Call Control service enhances the functionality of the Generic Call Control Service with leg management. It also allows for multi-party calls to be established, i.e., up to a service specific number of legs can be connected simultaneously to the same call.

The Multi-party Call Control Service is represented by the IpMultiPartyCallControlManager, IpMultiPartyCall, IpCallLeg 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 IpAppMultiPartyCallControlManager, IpAppMultiPartyCall and IpAppCallLeg to provide the callback mechanism.

6.1 Interface Class IpMultiPartyCallControlManager

Inherits from: IpService

This interface is the 'service manager' interface for the Multi-party Call Control Service. The multi-party call control manager interface provides the management functions to the multi-party call control service. The application programmer can use this interface to provide overload control functionality, create call objects and to enable or disable call-related event notifications. The action table associated with the STD shows in what state the IpMultiPartyCallControlManager must be if a method can successfully complete. In other words, if the IpMultiPartyCallControlManager is in another state the method will throw an exception immediately.

| <<Interface>> IpMultiPartyCallControlManager |
|--|
| createCall (appCall : in IpAppMultiPartyCallRef) : TpMultiPartyCallIdentifier createNotification (appCallControlManager : in IpAppMultiPartyCallControlManagerRef, notificationRequest : in TpCallNotificationRequest) : TpAssignmentID destroyNotification (assignmentID : in TpAssignmentID) : void changeNotification (assignmentID : in TpAssignmentID, notificationRequest : in TpCallNotificationRequest) : void <<deprecated>> getNotification () : TpNotificationRequestedSet setCallLoadControl (duration : in TpDuration, mechanism : in TpCallLoadControlMechanism, treatment : in TpCallTreatment, addressRange : in TpAddressRange) : TpAssignmentID <<new>> enableNotifications (appCallControlManager : in IpAppMultiPartyCallControlManagerRef) : TpAssignmentID <<new>> disableNotifications () : void <<new>> getNextNotification (reset : in TpBoolean) : TpNotificationRequestedSetEntry |

6.1.19 Method createNotification()

This method is used to enable call notifications so that events can be sent to the application. This is the first step an application has to do to get initial notifications of calls happening in the network. When such an event happens, the application will be informed by reportNotification(). In case the application is interested in other events during the context of a particular call session it has to use the createAndRouteCallLegReq() method on the call object or the eventReportReq() method on the call leg object. The application will get access to the call object when it receives the reportNotification(). (Note that createNotification() is not applicable if the call is setup by the application).

The createNotification method is purely intended for applications to indicate their interest to be notified when certain call events take place. It is possible to subscribe to a certain 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 or the specified criteria overlap with criteria already present in the network (when provisioned from within the network), the request is refused with P_INVALID_CRITERIA. The criteria are said to overlap when it leads to more than one application controlling the call or session at the same point in time during call or session processing.

If a notification is requested by an application with monitor mode set to notify, then there is no need to check the rest of the criteria for overlapping with any existing request as the notify mode does not allow control on a call to be passed over. Only one application can place an interrupt request if the criteria overlaps.

If a notification is requested by an application with an event type that is mutually exclusive compared to existing requested event types, then there is no need to check against the rest of the criteria for overlap. An example could be one application that trigger on “user busy” together with another application that trigger on “answer” – both requests should be allowed as only one can occur on the same call or session.

The overlap criteria have been defined to prevent multiple points of control, leading to possible interaction problems in networks that have no multi service support. Notice that dynamic aspects cannot be taken into account in the overlap criteria check. Therefore where dynamic event arming from an application causes a persistent control relationship it can prevent other applications to be invoked in the case single point of application control applies in the network.

However, the criteria check for overlap may as a network option be overruled by Multi Service networks allowing more services or applications to gain control of the same call or session at the same point in time. Refer to Call Control Common Definitions subpart of this specification (TS 29.198-4-1) for further details on application control over a call or session.

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 createNotification contains no callback, at the moment the application needs to be informed the gateway will use as callback the callback that has been registered by setCallback().

Returns assignmentID: Specifies the ID assigned by the call control manager interface for this newly-enabled event notification.