3GPP TSG CN Plenary Meeting #19 **NP-030030**
12- 14 March 2003, Birmingham, UK

Source: **CN5 (OSA)**

Title: **Rel-5 CRs 29.198-04-2 OSA API Part 4: Call control;**
Sub-part 2: Generic Call Control SCF

Agenda item: **8.2**

Document for: **APPROVAL**

Doc-1st-Level	Spec	CR	Rev	Phase	Subject	Cat	Version-Current	Doc-2nd-Level	Workitem
NP-030030	29.198-04-2	006	-	Rel-5	Correction of definition of the P_MAX_CALLLEGS_PER_CALL	F	5.1.0	N5-021149	OSA2

CR-Form-v5

CHANGE REQUEST

⌘ **29.198-04-2** **CR** **006** ⌘ **rev** **-** ⌘ Current version: **5.1.0** ⌘

*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 **X**

Title:	⌘	Correction of definition of the P_MAX_CALLLEGS_PER_CALL
Source:	⌘	N5
Work item code:	⌘	OSA2
Date:	⌘	15/10/2002
Category:	⌘	**F**
Release:	⌘	*REL-5*

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: ⌘ The P_MAX_CALLLEGS_PER_CALL property is defined for the multiparty call control service. The property "Indicates how many parties can be in one call" and is defined as being of the type INTEGER_SET.
The value of this property as used during registration for a camel phase 4 service is defined as {0,6}.
Because the properties can only be restricted during the creation of a profile, this would mean that only {}, {0}, {6} or {0,6} are possible values in a profile based on a camel phase 4 multiparty call control service.
Since it is desirable to limit the maximum parties that can used in one call this value should be changed to reflect all the possible values for the maximum number of parties that can be involved in the call.
Furthermore, it is doubtful what the use is of a maximum value of 0. This would indicate that the application would be able to create a call, but not create any parties in the call.
Additionally, it is unclear how the value of the property can be enforced on parties that are created by the network. E.g., what should the SCS do when the value of the property is 1 and an IDP on answer is received from the network.
Last, but not least, there might be situations where the SCS might want to limit the maximum number of active legs in a call for the network, but this can mean that there is 'temporary' one leg extra in the call in the SCS. E.g., in camel phase 2, there can only be two legs in a call in the network. However, restricting the application to two legs would mean that it would not be possible to create a follow-on call after disconnection of the B-leg, because this is done by creating a new leg, routing this new leg and then continueing the released B-leg leading to a call which temporary 3 legs. Therefore, we propose the rephrase the description of the property to "Indicates the maximum number of legs that represent an active connection to an end-point in the network. The enforcement of this property is only done when a leg is created or routed by the application"

Summary of change: ⌘ Change the value of the P_MAX_CALLLEGS_PER_CALL property for camel phase 4 multiparty callcontrol service to {1,2,3,4,5,6}
Rephrase the description of the property.

Consequences if not approved:	⌘	Not possible to create a useful profile/service level agreement for a multiparty call control service based on a camel phase4.

Clauses affected:	⌘			
Other specs affected:	⌘	☐ Other core specifications ☐ Test specifications ☐ 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/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.

Proposed Changes

8.1 List of Service Properties

The following table lists properties relevant for the MPCC API.

Property	Type	Description / Interpretation
P_TRIGGERING_EVENT_TYPES	INTEGER_SET	Indicates the static event types supported by the SCS. Static events are the events by which applications are initiated.
P_DYNAMIC_EVENT_TYPES	INTEGER_SET	Indicates the dynamic event types supported by the SCS. Dynamic events are the events the application can request for during the context of a call.
P_ADDRESSPLAN	INTEGER_SET	Indicates the supported address plan (defined in TpAddressPlan.) e.g. {P_ADDRESS_PLAN_E164, P_ADDRESS_PLAN_IP})
P_UI_CALL_BASED	BOOLEAN_SET	Value = TRUE : User interaction can be performed on call level and a reference to a Call object can be used in the IpUIManager.createUICall() operation. Value = FALSE: No User interaction on call level is supported.
P_UI_AT_ALL_STAGES	BOOLEAN_SET	Value = TRUE: User Interaction can be performed at any stage during a call . Value = FALSE: User Interaction can be performed in case there is only one party in the call.
P_MEDIA_TYPE	INTEGER_SET	Specifies the media type used by the Service. Values are defined by data-type TpMediaType : P_AUDIO, P_VIDEO, P_DATA
P_MAX_CALLLEGS_PER_CALL	INTEGER_SET	Indicates ~~how many~~ the maximum number of ~~parties~~ legs in a call for which a connection to a call party exists in the network~~that can be in one call.~~. The enforcement of this property is done only when a leg is created or routed by the application
P_UI_CALLLEG_BASED	BOOLEAN_SET	Value = TRUE : User interaction can be performed on leg level and a reference to a CallLeg object can be used in the IpUIManager.createUICall() operation. Value = FALSE : No user interaction on leg level is supported.
P_PARALLEL_INITIAL_ROUTING_REQUESTS	BOOLEAN_SET	Indicates whether for application initiated calls it is possible to issue multiple routing request methods in parallel or that the application has to wait for the result of the first request before another one can be invoked. Value = TRUE: Multiple routing requests can be invoked in parallel. Value = FALSE: Result of first request has to be received before another request can be issued.

The previous table lists properties related to capabilities of the SCS itself. The following table lists properties that are used in the context of the Service Level Agreement, e.g. to restrict the access of applications to the capabilities of the SCS.

Property	Type	Description
P_TRIGGERING_ADDRESSES	ADDRESS_RANGE_SET	Indicates for which numbers the notification may be set. For terminating notifications it applies to the terminating number, for originating notifications it applies only to the originating number. See further explanation on which events are originating and which are terminating, below.
P_MONITOR_MODE	INTEGER_SET	Indicates whether the application is allowed to monitor in interrupt and/or notify mode. Set is: P_INTERRUPT P_NOTIFY
P_NUMBERS_TO_BE_CHANGED	INTEGER_SET	Indicates which numbers the application is allowed to change or fill for legs in an incoming call. Allowed value set: {P_ORIGINAL_CALLED_PARTY_NUMBER, P_REDIRECTING_NUMBER, P_TARGET_NUMBER, P_CALLING_PARTY_NUMBER}.
P_CHARGEPLAN_ALLOWED	INTEGER_SET	Indicates which charging is allowed in the setCallChargePlan indicator. Allowed values: {P_TRANSPARANT_CHARGING, P_CHARGE_PLAN}
P_CHARGEPLAN_MAPPING	INTEGER_INTEGER_MAP	Indicates the mapping of chargeplans (we assume they can be indicated with integers) to a logical network chargeplan indicator. When the chargeplan supports indicates P_CHARGE_PLAN then only chargeplans in this mapping are allowed.
P_HIGH_PROBABILITY_OF_COMPLETION	BOOLEAN_SET	Value = TRUE : high probability of call completion field can be set. Value = FALSE : high probability of call completion field can not be set. FALSE is the default value.

The following table explains how the P_TRIGGERING_ADDRESSES property that is inherited via the Generic Call Control properties should be interpreted with respect to which of the notifications apply to originating numbers and which of the notifications apply to terminating numbers.

P_CALL_EVENT_ORIGINATING_CALL_ATTEMPT	Originating
P_CALL_EVENT_ORIGINATING_CALL_ATTEMPT_AUTHORISED	Originating
P_CALL_EVENT_ADDRESS_COLLECTED	Originating
P_CALL_EVENT_ADDRESS_ANALYSED	Originating
P_CALL_EVENT_ORIGINATING_SERVICE_CODE	Originating
P_CALL_EVENT_ORIGINATING_RELEASE	Originating
P_CALL_EVENT_TERMINATING_CALL_ATTEMPT	Terminating
P_CALL_EVENT_TERMINATING_CALL_ATTEMPT_AUTHORISED	Terminating
P_CALL_EVENT_ALERTING	Terminating
P_CALL_EVENT_ANSWER	Terminating
P_CALL_EVENT_TERMINATING_RELEASE	Terminating
P_CALL_EVENT_REDIRECTED	Terminating
P_CALL_EVENT_TERMINATING_SERVICE_CODE	Terminating
P_CALL_EVENT_QUEUED	N/A

8.2 Service Property values for the CAMEL Service Environment.

Implementations of the MultiParty Call Control API relying on the CSE of CAMEL phase 4 shall have the Service Properties outlined above set to the indicated values :

```
P_OPERATION_SET = {
“IpMultiPartyCallControlManager.createCall”,
“IpMultiPartyCallControlManager.createNotification”,
“IpMultiPartyCallControlManager.destroyNotification”,
“IpMultiPartyCallControlManager.changeNotification”,
“IpMultiPartyCallControlManager.getNotification”,
“IpMultiPartyCallControlManager.getNextNotification”,
```

```
"IpMultiPartyCallControlManager.enableNotifications",
"IpMultiPartyCallControlManager.disableNotifications",
"IpMultiPartyCallControlManager.setCallLoadControl"
"IpMultiPartyCall.getCallLegs",
"IpMultiPartyCall.createCallLeg",
"IpMultiPartyCall.createAndRouteCallLegReq",
"IpMultiPartyCall.release",
"IpMultiPartyCall.deassignCall",
"IpMultiPartyCall.getInfoReq",
"IpMultiPartyCall.setChargePlan",
"IpMultiPartyCall.setAdviceOfCharge",
"IpMultiPartyCall.superviseReq",
"IpCallLeg.routeReq",
"IpCallLeg.eventReportReq",
"IpCallLeg.release",
"IpCallLeg.getInfoReq",
"IpCallLeg.getCall",
"IpCallLeg.continueProcessing"
}

P_TRIGGERING_EVENT_TYPES = {
P_CALL_EVENT_ADDRESS_COLLECTED,
P_CALL_EVENT_ADDRESS_ANALYSED,
P_CALL_EVENT_ORIGINATING_RELEASE,
P_CALL_EVENT_TERMINATING_CALL_ATTEMPT_AUTHORISED,
P_CALL_EVENT_TERMINATING_RELEASE
}
```

Note: P_CALL_EVENT_ORIGINATING_RELEASE only for the routing failure case, TpReleaseCause = P_ROUTING_FAILURE

```
P_DYNAMIC_EVENT_TYPES = {
P_CALL_EVENT_ALERTING,
P_CALL_EVENT_ANSWER,
P_CALL_EVENT_ORIGINATING_RELEASE,
P_CALL_EVENT_ORIGINATING_SERVICE_CODE,
P_CALL_EVENT_TERMINATING_RELEASE,
P_CALL_EVENT_TERMINATING_SERVICE_CODE
}

P_ADDRESS_PLAN = {
P_ADDRESS_PLAN_E164
}

P_UI_CALL_BASED = {
TRUE
}

P_UI_AT_ALL_STAGES = {
FALSE
}

P_MEDIA_TYPE = {
P_AUDIO
}

P_MAX_CALLLEGS_PER_CALL = {
0,
1,
2,
3,
4,
5,
6
}

P_UI_CALLLEG_BASED = {
TRUE
}

P_MEDIA_ATTACH_EXPLICIT = {
FALSE
}
```