ETSI TR 102 397-3-1 V1.1.1 (2005-08)

Technical Report

Open Service Access (OSA); Mapping of Parlay X Web Services to Parlay/OSA APIs; Part 3: Call Notification Mapping; Sub-part 1: Mapping to Generic Call Control

Reference

DTR/TISPAN-01021-03-01-OSA

Keywords

API, OSA, service

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Copyright Notification

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN).

The present document is part 3, sub-part 1 of a multi-part deliverable covering Open Service Access (OSA); Mapping of Parlay X Web Services to Parlay/OSA APIs, as identified below:

Part 1: "Common Mapping";

Part 2: "Third Party Call Mapping";

Part 3: "Call Notification Mapping";

Sub-part 1: "Mapping to Generic Call Control";

Sub-part 2: "Mapping to Multi-Party Call Control";

Part 4: "Short Messaging Mapping";

Part 5: "Multimedia Messaging Mapping";

Part 6: "Payment Mapping";

Part 7: "Account Management Mapping";

Part 8: "Terminal Status Mapping";

Part 9: "Terminal Location Mapping";

Part 10: "Call Handling Mapping";

Part 11: "Audio Call Mapping";

Part 12: "Multimedia Conference Mapping";

Part 14: "Presence Mapping".

NOTE: Part 13 has not been provided as there is currently no defined mapping between ES 202 391-13 [4] and the Parlay/OSA APIs. If a mapping is developed, it will become part 13 of this series.

The present document has been defined jointly between ETSI, The Parlay Group (http://www.parlay.org) and the 3GPP.

1 Scope

The present document specifies the mapping of the Parlay X Call Notification Web Service to the Parlay/OSA Generic Call Control Service Capability Feature (SCF).

The Parlay X Web Services provide powerful yet simple, highly abstracted, imaginative, telecommunications functions that application developers and the IT community can both quickly comprehend and use to generate new, innovative applications.

The Open Service Access (OSA) specifications define an architecture that enables application developers to make use of network functionality through an open standardized interface, i.e. the Parlay/OSA APIs.

2 References

For the purposes of this Technical Report (TR), the following references apply:

[1] ETSI TR 121 905: "Universal Mobile Telecommunications System (UMTS); Vocabulary for 3GPP Specifications (3GPP TR 21.905)".

[2] W3C Recommendation (2 May 2001): "XML Schema Part 2: Datatypes".

NOTE: Available at http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/.

[3] ETSI TR 102 397-1: "Open Service Access (OSA); Mapping of Parlay X Web Services to Parlay/OSA APIs; Part 1: Common Mapping".

[4] ETSI ES 202 391-13: "Open Service Access (OSA); Parlay X Web Services; Part 13: Address List Management".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TR 102 397-1 [3] apply.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 102 397-1 [3] apply.

4 Mapping description

The Call Notification capability can be implemented with the Parlay/OSA Generic Call Control SCF.

It is applicable to ETSI OSA 1.x/2.x/3.x, Parlay/OSA 3.x/4.x/5.x and 3GPP Releases 4 to 6.

5 Sequence diagrams

5.1 Notification of a network-related call event

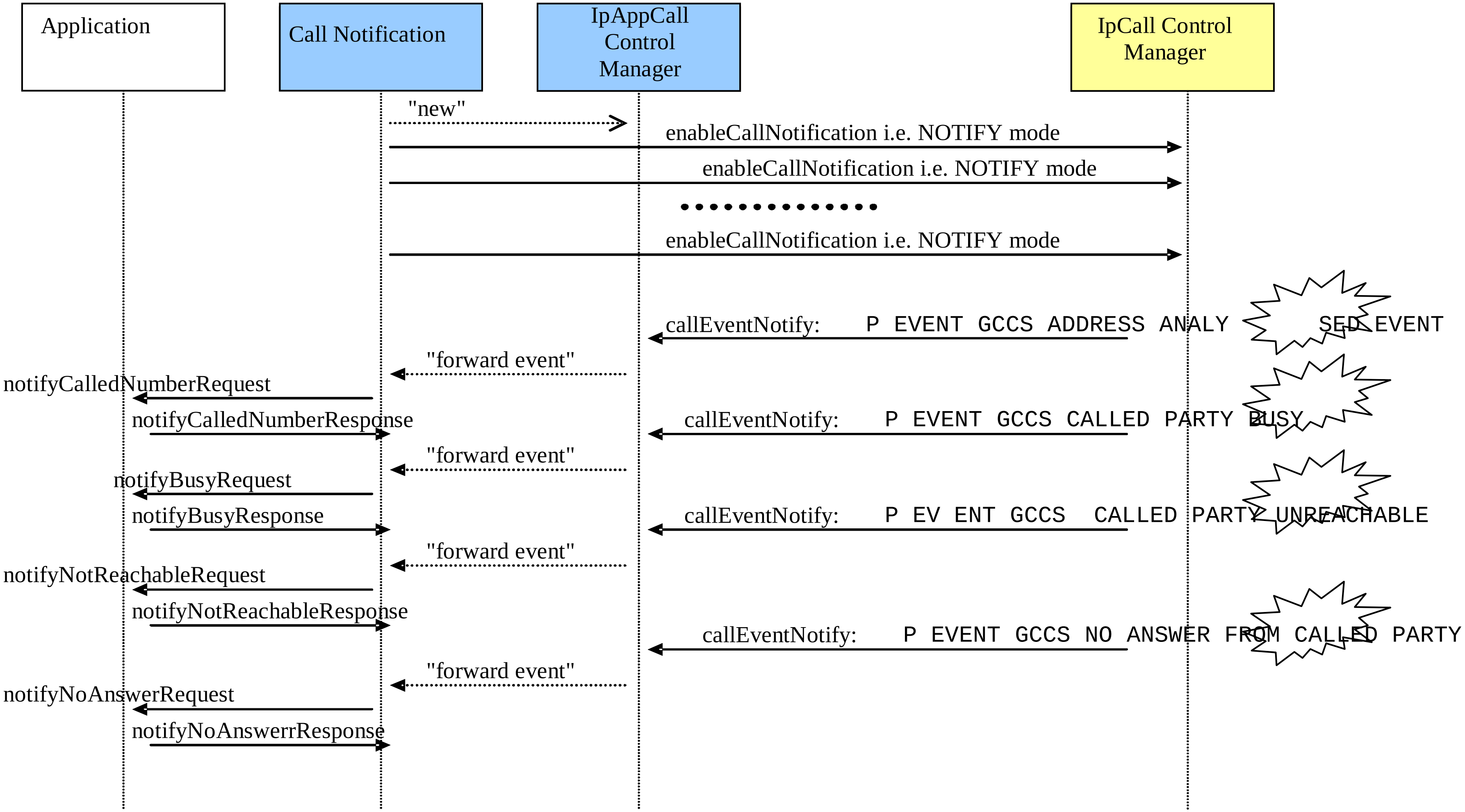
Application
Call Notification
IpAppCall Control Manager
IpCall Control Manager
"new"
enableCallNotification i.e. NOTIFY mode
enableCallNotification i.e. NOTIFY mode
enableCallNotification i.e. NOTIFY mode
callEventNotify: P_EVENT_GCCS_ADDRESS_ANALYSED_EVENT
"forward event"
notifyCalledNumberRequest
notifyCalledNumberResponse
callEventNotify: P_EVENT_GCCS_CALLED_PARTY_BUSY
"forward event"
notifyBusyRequest
notifyBusyResponse
callEventNotify: P_EV ENT_GCCS_ CALLED_PARTY_UNREACHABLE
"forward event"
notifyNotReachableRequest
notifyNotReachableResponse
callEventNotify: P_EVENT_GCCS_NO_ANSWER_FROM_CALLED_PARTY
"forward event"
notifyNoAnswerRequest
notifyNoAnswerrResponse

Figure 1

5.2 Handling a network-related call event - Action: Route

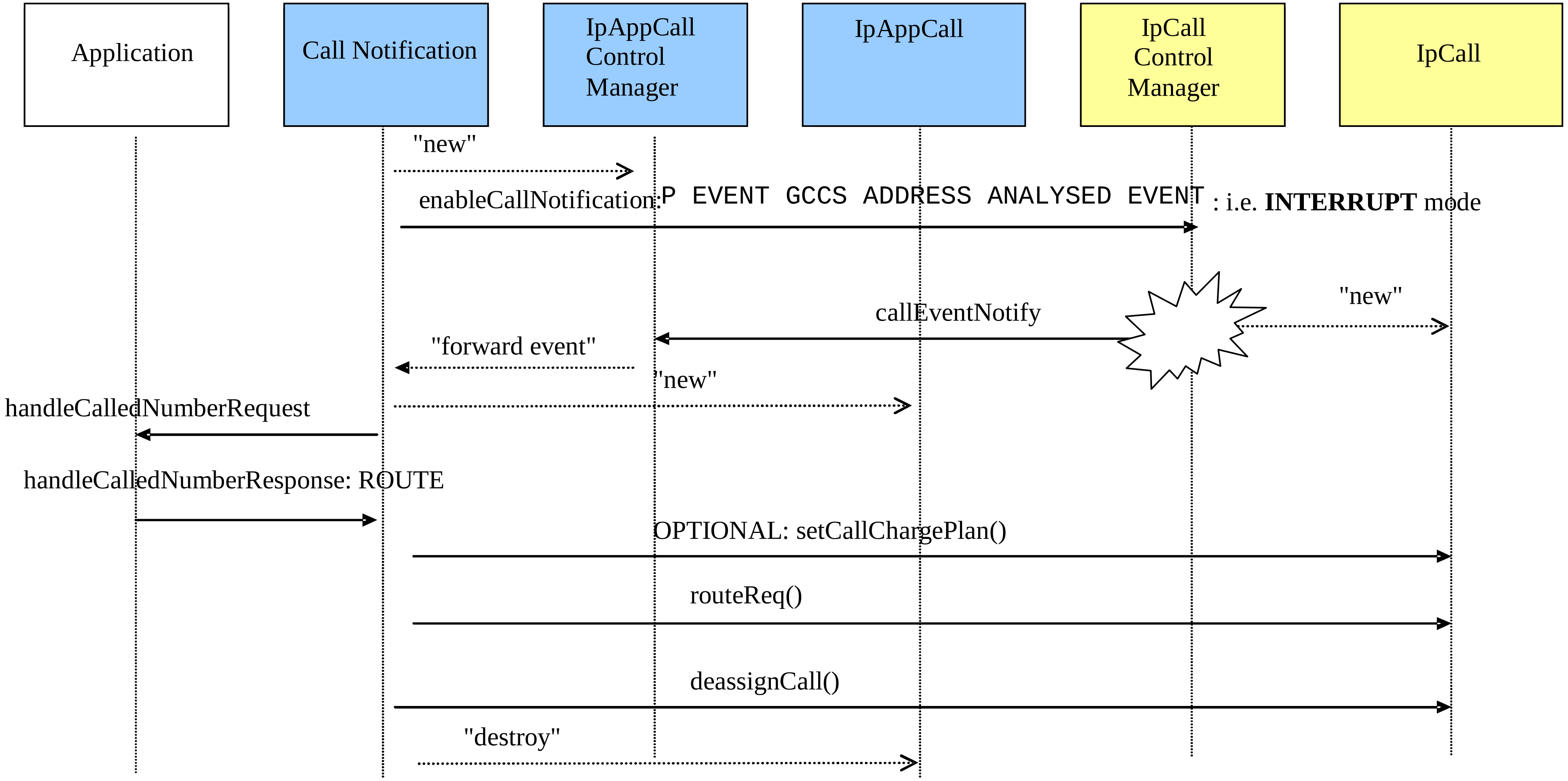
Application
Call Notification
IpAppCall Control Manager
IpAppCall
IpCall Control Manager
IpCall
"new"
enableCallNotification:P_EVENT_GCCS_ADDRESS_ANALYSED_EVENT : i.e. INTERRUPT mode
callEventNotify
"new"
"forward event"
"new"
handleCalledNumberRequest
handleCalledNumberResponse: ROUTE
OPTIONAL: setCallChargePlan()
routeReq()
deassignCall()
"destroy"

Figure 2

5.3 Handling a network-related call event - Action: Continue

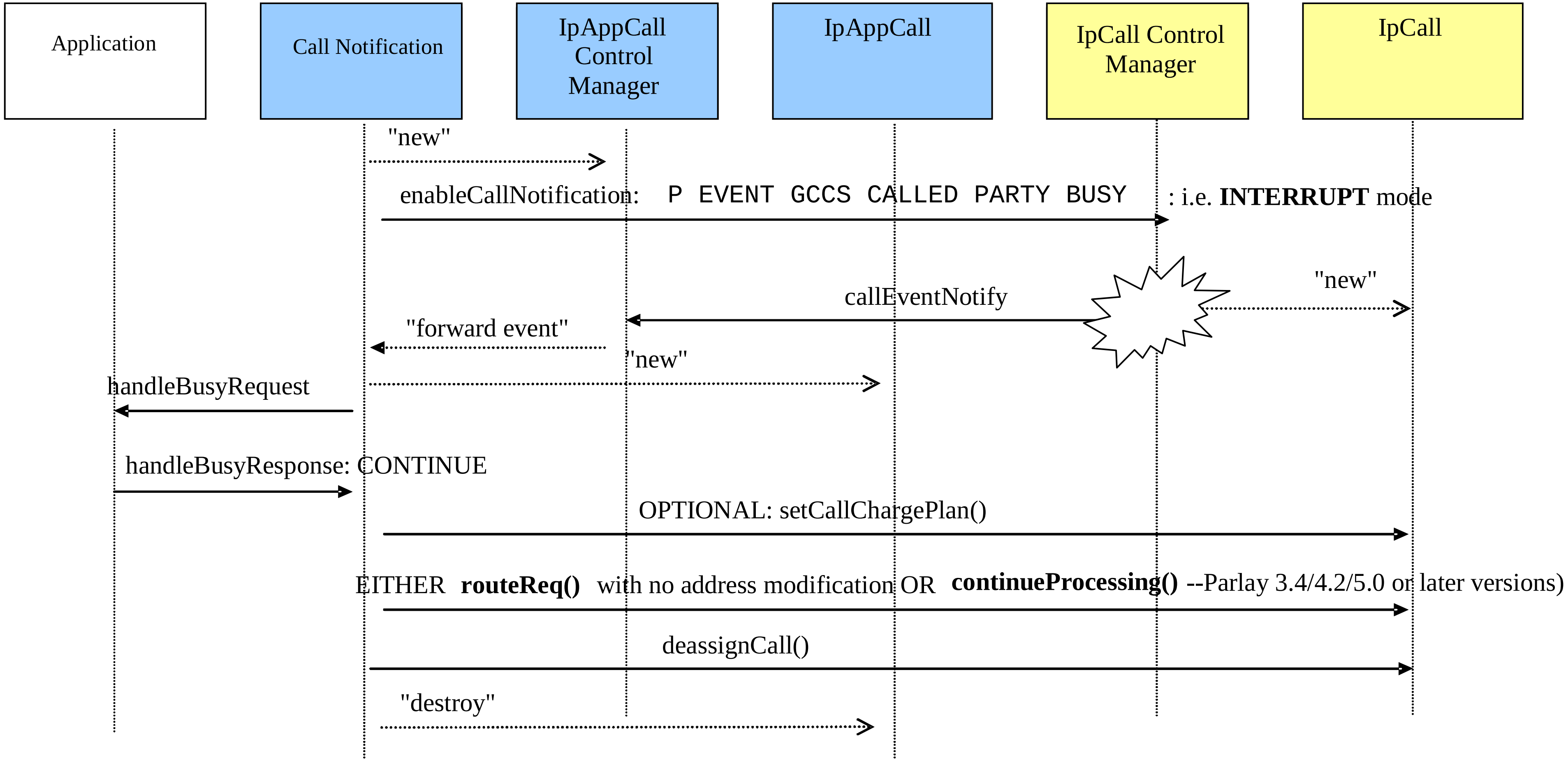
Application
Call Notification
IpAppCall Control Manager
IpAppCall
IpCall Control Manager
IpCall
"new"
enableCallNotification: P_EVENT_GCCS_CALLED_PARTY_BUSY : i.e. INTERRUPT mode
callEventNotify
"new"
"forward event"
"new"
handleBusyRequest
handleBusyResponse: CONTINUE
OPTIONAL: setCallChargePlan()
EITHER routeReq() with no address modification OR continueProcessing() --Parlay 3.4/4.2/5.0 or later versions)
deassignCall()
"destroy"

Figure 3

5.4 Handling a network-related call event - Action: End Call

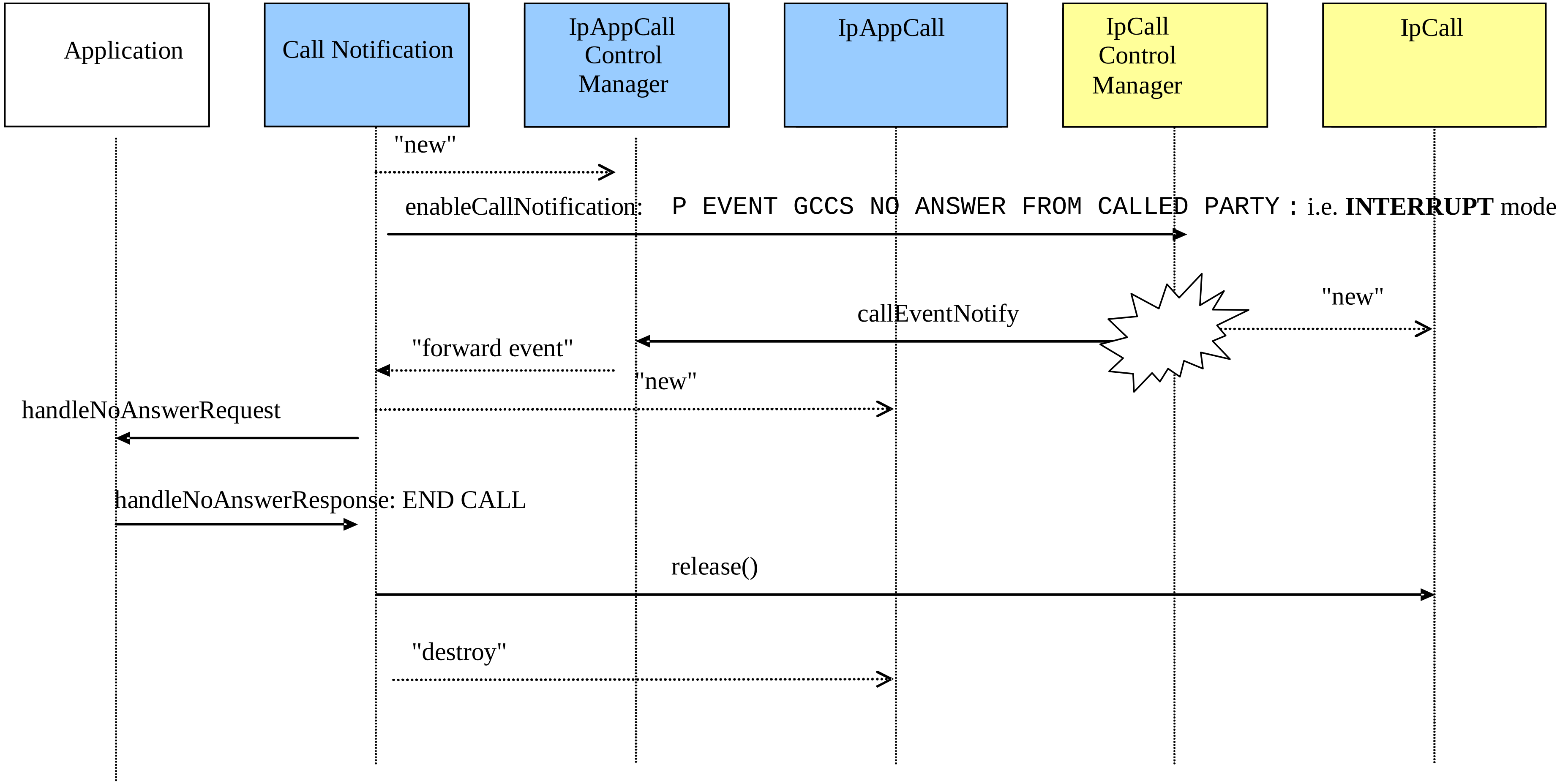

Figure 4

6 Detailed mapping information

6.1 Operations

6.1.1 notifyXXX

The sequence diagram in clause 5.1 illustrates the flow for any of the **notifyXXX** operations, namely:

- notifyCalledNumber;
- notifyBusy;
- notifyNotReachable;
- notifyNoAnswer.

Each **notifyXXX** operation is mapped from the Parlay/OSA method:
IpAppCallControlManager.callEventNotify.

6.1.1.1 Mapping from IpAppCallControlManager.callEventNotify

The IpAppCallControlManager.callEventNotify method is invoked with the following parameters.

Name	Type	Comment
callReference	TpCallIdentifier	Specifies the reference to the call interface to which the notification relates. Since the notification is in NOTIFY mode, this parameter should be ignored by the Call Notification web service; and is is not mapped to the **notifyXXXRequest** message.
eventInfo	TpCallEventInfo	Specifies event-related data, which is mapped to the **notifyXXXRequest** message as described in clause 6.1.1.2.
assignmentID	TpAssignmentID	Specifies the assignment id returned after an earlier invocation of IpAppCallControlManager.enableCallNotification method, when the criteria associated with this call-related event were activated in the network. It is used internally by the Call Notification web service to correlate the Parlay/OSA callbacks.

The result from IpAppCallControlManager.callEventNotify is of type IpAppCallRef, but is null for event notifications in NOTIFY mode.

6.1.1.2 Mapping from `TpCallEventInfo` to notifyXXXRequest

The elements of the `TpCallEventInfo` data type are mapped to the parts of the **notifyXXXRequest** message as follows.

Name	Type	Comment
DestinationAddress	TpAddress	Mapped to the URI provided in the **CalledParty** part of **notifyXXXRequest**, as described in TR 102 397-1 [3].
OriginatingAddress	TpAddress	Mapped to the URI provided in the **CallingParty** part of **notifyXXXRequest**, as described in TR 102 397-1 [3].
OriginalDestinationAddress	TpAddress	Not mapped.
RedirectingAddress	TpAddress	Not mapped.
CallAppInfo	TpCallAppInfoSet	Not mapped.
CallEventName	TpCallEventName	Defines the specific **notifyXXXRequest** message to send, i.e.: **XXX**= **CalledNumber** for `P_EVENT_GCCS_ADDRESS_ANALYSED_EVENT;` **XXX**= **Busy** for `P_EVENT_GCCS_CALLED_PARTY_BUSY;` **XXX**= **NotReachable** for `P_EVENT_GCCS_ CALLED_PARTY_UNREACHABLE;` **XXX**= **NoAnswer** for `P_EVENT_GCCS_NO_ANSWER_FROM_CALLED_PARTY.`
CallNotificationType	TpCallNotificationType	Not mapped.
MonitorMode	TpCallMonitorMode	Not mapped. This element has a value of `P_CALL_MONITOR_MODE_NOTIFY`.

6.1.2 handleXXX

The sequence diagrams in clauses 5.2 through 5.4 illustrates the flow for any of the **handleXXX** operations.

A **handleXXX** operation is synchronous from the Call Notification Web Service's point of view.

As defined for the **notifyXXXrequest** message in clause 6.1.1, each **handleXXXrequest** message is mapped from the Parlay/OSA method: `IpAppCallControlManager.callEventNotify`. However in this case the `eventInfo.MonitorMode` element has a value of `P_CALL_MONITOR_MODE_INTERRUPT`.

The **handleXXXresponse** message is mapped to one or more of the following Parlay/OSA methods, depending on the value of the **Action** part:

- `IpCall.routeReq`, if the **ActionToPerform** parameter of the **Action** part = **Route or Continue**. (For **ActionToPerform** = **Continue,** an alternative to `routeReq` is `continueProcessing`. However, this alternative is only available in Parlay 3.4/4.2/5.0 and later versions).
- `IpCall.setCallChargePlan`, if:
  - the optional **Charging** parameter of the **Action** part is present; and
  - the **ActionToPerform** parameter of the **Action** part = **Route or Continue.**
- `IpCall.continueProcessing` (Parlay 3.4/4.2/5.0 and later versions only), if the **ActionToPerform** parameter of the **Action** part = **Continue.**
- `IpCall.deassignCall`, if the **ActionToPerform** parameter of the **Action** part = **Route or Continue.**
- `IpCall.release`, if the **ActionToPerform** parameter of the **Action** part = **EndCall.**

6.1.2.1 Mapping to IpCall.routeReq

The IpCall.routeReq method is invoked with the following parameters.

Name	Type	Comment
callSessionID	TpSessionID	Not mapped: derived from IpAppCallControlManager.callEventNotify.
responseRequested	TpCallReportRequestSet	Not mapped. [Requests NO call-related event reports].
targetAddress	TpAddress	Specifies the destination leg to which the call should be routed. In the case where the **ActionToPerform** = **Route,** it is constructed based on the URI provided in the **RoutingAddress** parameter of the **Action** part of the **handleXXXResponse** message, mapped as described in TR 102 397-1 [3]. In the case where the **ActionToPerform** = **Continue**, it is the same as the eventInfo.DestinationAddress parameter provided in the IpAppCallControlManager.callEventNotify.
originatingAddress	TpAddress	Not mapped: derived from IpAppCallControlManager.callEventNotify.
originalDestination Address	TpAddress	Not mapped: derived from IpAppCallControlManager.callEventNotify.
redirectingAddress	TpAddress	Not mapped: derived from IpAppCallControlManager.callEventNotify.
appInfo	TpCallAppInfoSet	Not mapped: derived from IpAppCallControlManager.callEventNotify.

The result from IpCall.routeReq is of type TpSessionID and is not mapped to the Parlay X interface.

Parlay exceptions thrown by IpCall.routeReq are not mapped to Parlay X exceptions.

6.1.2.2 Mapping to IpCall.setCallChargePlan

The IpCall.setCallChargePlan method is invoked with the following parameters.

Name	Type	Comment
callSessionID	TpSessionID	Not mapped: derived from IpAppCallControlManager.callEventNotify.
callChargePlan	TpCallChargePlan	Specifies the charge plan to use. It is constructed based on the values provided in the optional **Charging** parameter of the **Action** part of the **handleXXXResponse** message. See the following table for details.

The callChargePlan parameter is constructed as follows.

Name	Type	Comment
ChargeOrderType	TpCallChargeOrderCategory	Not mapped.
TransparentCharge	TpOctetSet	Specifies an operator-specific charge plan. It is constructed using the value of the **ChargingInformation.contract** element provided in the **Charging** parameter.
ChargePlan	TpInt32	Not mapped.
AdditionalInfo	TpOctetSet	Descriptive string sent to billing system. It is constructed using the value of the **ChargingInformation.description** element provided in the **Charging** parameter. (May optionally include values of other elements of the **Charging** parameter.)
PartyToCharge	TpCallPartyToChargeType	Not mapped.
PartyToChargeAdditionalInfo	TpCallPartyToChargeAdditional Info	Not mapped.

Parlay exceptions thrown by IpCall.setCallChargePlan are not mapped to Parlay X exceptions.

6.1.2.3 Mapping to `IpCall.continueProcessing` (Parlay 3.4/4.2/5.0)

The `IpCall.continueProcessing` method is invoked with the following parameters.

Name	Type	Comment
callSessionID	TpSessionID	Not mapped: derived from `IpAppCallControlManager.callEventNotify.`

Parlay exceptions thrown by `IpCall.continueProcessing` are not mapped to Parlay X exceptions.

6.1.2.4 Mapping to `IpCall.deassignCall`

The `IpCall.deassignCall` method is invoked with the following parameters.

Name	Type	Comment
callSessionID	TpSessionID	Not mapped: derived from `IpAppCallControlManager.callEventNotify.`

Parlay exceptions thrown by `IpCall.deassignCall` are not mapped to Parlay X exceptions.

6.1.2.5 Mapping to `IpCall.release`

The `IpCall.release` method is invoked with the following parameters.

Name	Type	Comment
callSessionID	TpSessionID	Not mapped: derived from `IpAppCallControlManager.callEventNotify.`
cause	TpCallReleaseCause	Not mapped. Value should indicate application-directed termination of the call attampt.

Parlay exceptions thrown by `IpCall.release` are not mapped to Parlay X exceptions.

6.2 Exceptions

For the present document, the mapping of Parlay/OSA API method exceptions to Parlay X Web Service exceptions is null.

7 Additional notes

No additional notes are provided.

History

Document history		
V1.1.1	August 2005	Publication