O2 Czech Republic a.s.	Technical specification external	TE000002
Validity date:	05.09.2018	Version: 08.00
Expiration date:		Page 1 z 46
Security classification:	SEC-C0 (Veřejné)	

TE000002

Interface for VoIP Connect Services

Scope:

The document specifies the interface for VoIP Connect

Applicability:

The document is obligatory for the O2 Czech Republic a.s. company.

Process:

Technologie hlasových a mob.sítí a služeb

Document Garant		Process Owner		Approbator	
Poupě Petr		Čapek Jiří			
Date	*Signature*	*Date*	*Signature*	*Date*	*Signature*

O2 Czech Republic a.s.,
registration no. (IČ) 601 93 336,
registerid in the Commercial Registry of the City Court in Prague,section B, insert 2322
adress Za Brumlovkou 266/2, 140 22 Praha 4 - Michle, Czech Republic

TABLE OF CONTENT:

1 Initial provisions

1.1 Scope

The document specifies an end user interface for VoIP Platform Voice over IP services of O2 Czech Republic a.s.

1.2 Validity and obligation

This document is obligatory for the company O2 Czech Republic a.s. It is issued for the general public use, especially for the suppliers and producers of CPEs. It is approved in Czech and English version. In case of a different interpretation the Czech version is to be used. It is valid from the date set on the first page.

1.3 Document history

Ver.	Datum	Název	Poznámka
3	04/2009	Rozhraní pro služby VoIP Connect	Převedení z P8 ver 3,5
4	10/2010	Rozhraní pro služby VoIP Connect	Revize dokumentu
5	10/2012	Rozhraní pro služby VoIP přes IMS	Revize dokumentu
6	4/2013	Rozhraní pro služby VoIP přes IMS	Revize dokumentu
7	6/2014	Rozhraní pro služby VoIP přes IMS	Revize dokumentu - Doplnění kap.4 TECHNICAL SPECIFICATION OF SIP UNI VOIP END USER INTERFACE – specific interface for Fixed Network Voice Service Replacement.
8.	3/2015	Rozhraní pro služby VoIP přes IMS	Revize dokumentu - oprava názvu společnosti
9.	09/2018	Rozhraní pro služby VoIP přes IMS	Kontrola a revize dokumentu.

1.4 Definitions

For definitions see the text.

1.5 Abbreviations

For abbreviations see the text.

1.6 Records

This document does not require records in the sense of the Directive SM000594, it has the nature of a recommendation and/or technical information.

1.7 Reference

The document stands in connection with other published Technical Specifications (TE) of the O2 Czech Republic a.s. company.

2 Supported interfaces and codecs

2.1 Supported interfaces

O2 Czech Republic a.s. provides following VoIP Platform Voice over IP interfaces
- SIP UNI VOIP END USER INTERFACE – with user registration
- SIP NNI VOIP END USER INTERFACE
  – without VoIP Centrex Support and without registration
  – with VoIP Centrex Support and with registration

IP interface is physically connected via 802.3 Ethernet, RJ-45 connector

Additionaly there are provided several types of CPGs (Customer premises Gateways), that simulate the traditional PSTN interfaces:
POTS (analogue Z interface)
ISDN BRI
ISDN PRI

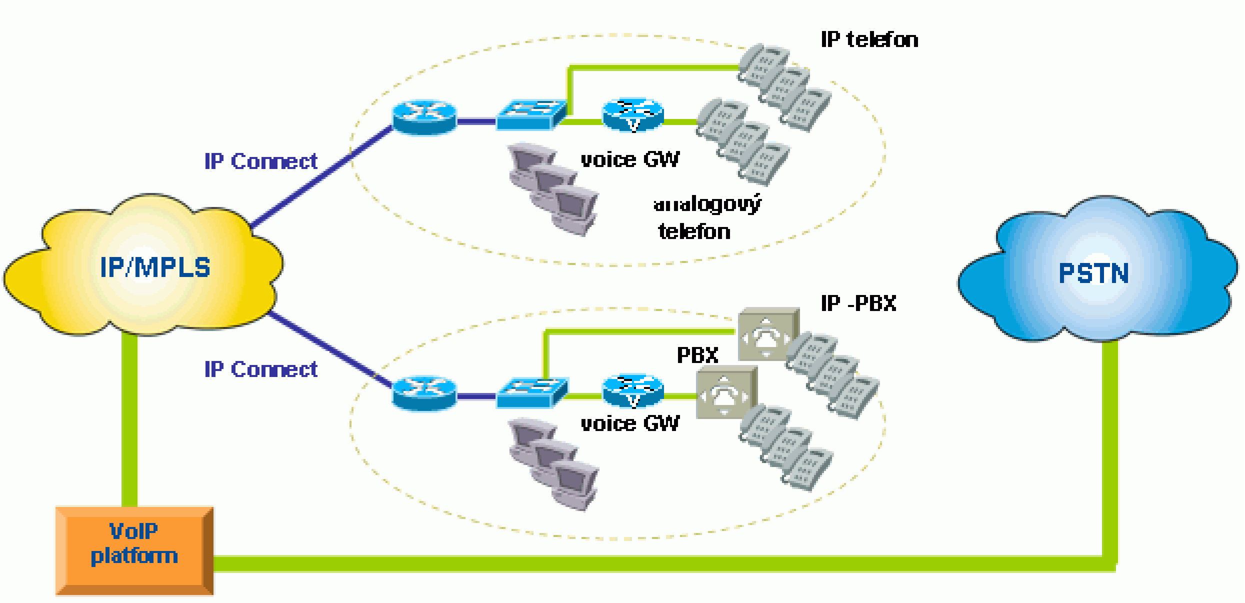
2.2 Supported codecs

Voice service: G.729, G.711, G.722.2
Modem transmission service: G.711
Fax transmission service: T.38 (+G.711)
Circuit mode 64kbit/s unrestricted (ISDN): Clear Mode

2.3 Codec selection method

Codec selection has to be in accordance with RFC 3264. For Voice connection only codec G.729 shall be offered in INVITE (SDP part) message, in case of modem or fax tone detection the Re-Invite method shall be used for new codec negotiation with codecs T.38/G.711 for fax and G.711 pro modem communication).

In case of Circuit mode 64kbit/s unrestricted requirement only Clear Mode codec shall be offered in INVITE (SDP part) message.



3 TECHNICAL SPECIFICATION OF SIP UNI VOIP END USER INTERFACE – WITH USER REGISTRATION

Specification of VoIP protocols for single users with registration to SIP server.
Specifikation is relevant for SIP phones and for Customer premises GW (CPG) with analogue phone user interface Z and ISDN BRI interface.

3.1 List of standards applied to end user interface definition

RFC 1889: RTP: A Transport Protocol for Real-Time Applications, January 1996
RFC 1890: RTP Profile for Audio and Video Conferences with Minimal Control, January 1996
RFC 2327: SDP: Session Description Protocol, April, 1998
RFC 2806: URLs for Telephone Calls, April, 2000
RFC 4733: RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals, 2006
RFC 3261: SIP: Session Initiation Protocol, June 2002
RFC 3262: Reliability of Provisional Responses in SIP, June 2002
RFC 3263: Session Initiation Protocol (SIP): Locating SIP Servers, June 2002
RFC 3264: An Offer/Answer Model with the Session Description Protocol, June 2002
RFC 3265: Session Initiation Protocol (SIP)-Specific Event Notification, June 2002
RFC 3266: Support for IPv6 in Session Description Protocol (SDP), June 2002
RFC 3311: The Session Initiation Protocol (SIP) UPDATE Method, September 2002
RFC 3323: A Privacy Mechanism for the Session Initiation Protocol (SIP), November 2002
RFC 3325: Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks, November 2002
RFC 3326: The Reason Header Field for the Session Initiation Protocol (SIP)", RFC 3326, December 2002
RFC 3428: Session Initiation Protocol (SIP) Extension for Instant Messaging, December 2002
RFC 3455: Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the Third-Generation Partnership Project (3GPP), January 2003
RFC 3515: The Session Initiation Protocol (SIP) Refer Method, April, 2003
RFC 3550: RTP: A Transport Protocol for Real-Time Applications, July 2003
RFC 3551: RTP Profile for Audio and Video Conferences with Minimal Control, July 2003

RFC 3725: Best Current Practices for Third Party Call Control (3pcc) in the Session Initiation Protocol (SIP), April, 2004
RFC 3842: A Message Summary and Message Waiting Indication Event Package for the Session Initiation Protocol (SIP), August 2004
RFC 3891: The Session Initiation Protocol (SIP) "Replaces" Header, September 2004
RFC 3892: The Session Initiation Protocol (SIP) Referred-By Mechanism, September 2004
3GPP TS 24.647 v8.0.0 Advice of Charge (AoC)

3.2 Session Initiation Protocol - RFC 3261: Session Initiation Protocol, June 2002

3.2.1 Support of Authentication

VoIP Platform supports initiating challenges as well as responding to challenges from devices on the access interface.
VoIP Platform supports sending and receiving SIP digest MD5 authentication challenges only.
The following clarifications are provided for VoIP Platform authentication support:
The VoIP Platform Application Server challenges REGISTER, INVITE, REFER, and UPDATE requests. These requests are only challenged if the user has the Authentication service assigned and enabled.
− Note, the REGISTER and REFER requests are always challenged if the Authentication service is assigned and enabled.
− The INVITE request is challenged based on a configurable percentage. Re-INVITE challenges may be configured independently of INVITE challenges.
− The UPDATE request is challenged only if it contains a session descriptor or if it specifies a contact change. The UPDATE request is challenged similarly to a re-INVITE request based on a configurable percentage.

3.2.2 Support of the OPTIONS Method

VoIP Platform uses the OPTIONS method to determine connectivity of devices on the access interface. It is used as an application-layer ping to detect SIP responsiveness of the device.
Access devices must support receiving the OPTIONS method. The access device must provide a SIP response to the OPTIONS method. However, the access device does not have to respond with a 200 OK to the options method. The device may respond with any SIP response code, although a 200 OK is preferred.
Note that VoIP Platform does not include SDP capabilities in the OPTIONS request sent to devices.
VoIP Platform supports receiving the OPTIONS request and responds with a 200 OK. Note that VoIP Platform does not include SDP capabilities in the OPTIONS response.

3.2.3 SIP transport protocol

UDP shall be used as SIP transport protocol

3.2.4 SIP timers

VoIP Platform implements the various SIP timers defined in RFC 3261. The following table describes VoIP Platform-specific behavior.

Timer	Value/Comment
T1	500ms
T2	4 seconds
T4	Not used (see Timer I and Timer K in this table).
Timer A	Timer A is used by VoIP Platform as defined in RFC 3261. It is initialized to T1 and doubles at every retransmission.

Timer D	The wait time for late retransmission is handled differently in VoIP Platform. Received messages are remembered by VoIP Platform and kept in memory until the stale message audit is run. This audit is run at most once every 10*T2. When it executes, messages older than 10*T2 are removed. This is equivalent to Timer D firing after an interval of at least 10*T2, but possibly 20*T2 or more, for both UDP and TCP. Furthermore, this time can be shortened if the SIP dialog is released. If useSessionCompletionTimer is true, the dialog post-released retention period is configured using sessionCompletionTimer value (5 seconds to 100 seconds). If useSessionCompletionTimer is false, the dialog post-released retention period is set to “10*T2”.
Timer E	Timer E is used by VoIP Platform as defined in RFC 3261. It is initialized to T1 and doubles at every retransmission until beginning to use T2 as the interval. VoIP Platform does not start using interval T2 upon receiving a 1xx response; the RFC 3261 non-compliance is to accommodate devices that are not compliant with RFC 4320. When VoIP Platform is in Yellow CallP Overload condition, Timer E is initialized to 2*T1. When VoIP Platform is in Red CallP Overload condition, Timer E is initialized to 4*T1.
Timer G	Timer G is used by VoIP Platform as defined in RFC 3261. It is initialized to T1 and doubles at every retransmission. However, VoIP Platform does not use T2 as the maximum interval between retransmissions. VoIP Platform also uses this timer for retransmitting INVITE 2xx responses and reliable 1xx responses.
Timer I	The wait time for late retransmission is handled differently in VoIP Platform. See comment for Timer D.
Timer J	The wait time for late retransmission is handled differently in VoIP Platform. See comment for Timer D.

Timer K	The wait time for late retransmission is handled differently in VoIP Platform. See comment for Timer D.

3.2.5 SIP Subscriber Identification/Addressing

SIP subscriber identification/addressing:
URLs for Telephone Calls (RFC 2806)
The tel URI for Telephone Numbers (RFC 3966)

The calling party's category tel URI parameter (draft-mahy-iptel-cpc-00)
A Privacy Mechanism for the Session Initiation Protocol (SIP) (RFC 3323)
Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks (RFC 3325)
Diversion Indication in SIP (RFC 5806)
VoIP Platform supports all standard SIP functionality for addressing, as specified in RFC 3261. VoIP Platform also supports a number of specifications related to subscriber identification and addressing in SIP. Highlights and clarifications of this support are provided below:
Addressing Information:
− VoIP Platform can (optionally) restrict the calling party identity upon receipt of anonymous in the display name of the From header. When VoIP Platform receives “anonymous” in the display name of the From header from an access device, VoIP Platform treats the calling party identity as restricted and does not pass the calling party identity out to any untrusted entities. VoIP Platform also supports other privacy specifications described below on the access interface, which can also restrict the calling party identity.
− For calls with restricted calling party identity to be sent to an access device from VoIP Platform, VoIP Platform inserts "Anonymous" <sip:anonymous@anonymous.invalid> in the From header to honor the calling party identity blocking request.
− VoIP Platform also supports called party identity including name and number. VoIP Platform supports both Tel URIs and SIP URIs. VoIP Platform also supports the telephone-subscriber contained in the SIP URI including all proper escaping.
− Access devices do not need to send anything other than the FROM header in INVITEs to identify the device/subscriber. VoIP Platform accepts INVITEs from the device with Remote-Party-ID or P-Asserted-Identity and P-Preferred-Identity headers. However, it is recommended that devices send only the FROM header since the other headers provide no functional benefit to the FROM header on the access interface.
− In cases where the identification headers are not identical, VoIP Platform uses the following rule of precedence to determine the identity of the subscriber:
P-Preferred-Identity if present, else P-Asserted-Identity if present, else Remote-Party-ID if present, else From
− If the Application Server finds a match for a P-Asserted-Identity header for a VoIP Platform subscriber, then that subscriber is the originator. If the P-Asserted-Identity header has both SIP URI and a TEL URI, then the Application Server gives precedence to the SIP URI. If the SIP URI has a user=phone, then the Application Server tries to find a subscriber with a matching DN. If the P-Asserted-Identity has both SIP URI and TEL URI, and if the Application Server cannot find a subscriber that matches the SIP URI, but it can find a subscriber that matches the TEL URI, then it accepts that matched subscriber as the originator.
− If the Application Server cannot find a match for a P-Asserted-Identity header, then it tries to find a subscriber that matches the From header.

For INVITES sent from a VoIP Platform server:

Header	Format	Parameter	Value
Request-URI	SIP URI	User	Access device user name/address of record/line/port.
		Host	Access device domain name or IP address as specified in device inventory or registered contact (for devices which register).
From		display-name	User or group name, if

			available
	SIP URI	User	User or group phone number. Number is in national number format.
		Host	Application Server cluster domain name, Subscriber domain name, or IP address.
To	SIP URI	User	Access device user name/address of record/line/port.
		Host	Access device domain name or IP address as specified in device inventory or registered contact (for devices which register).
Contact	SIP URI	User	Not used (empty).
		Host	Application Server cluster domain name or IP address.

For INVITES sent from an Access Device:

Header	Format	Parameter	Value
Request-URI	SIP URI	User	Dialed number/national number, E.164 number, or dialed number with global or local network prefix phone-context.
		Host	Application Server cluster domain name, Subscriber domain name, Application Server alias, or IP address.
	TEL URI		Dialed number/national number, E.164 number, or dialed number with global or local network prefix phone-context.
From		display-name	Calling party name, if available.
	SIP URI	User	Access device user name/address of record/line/port.
		Host	Access device domain name or IP address or Application Server cluster domain name, Subscriber domain name, Application Server alias, or Application Server IP address (for devices which register).

To	SIP URI	User	Should be dialed number/national number, E.164 number, or dialed number with global or local network prefix phone-context. However, VoIP Platform does not perform any translations on the To header SIP URI. Therefore, the To header SIP URI may contain anything.
		Host	Application Server cluster domain name, Subscriber domain name, Application Server alias, or IP address.
	Tel URI		Should be dialed number/national number, E.164 number, or dialed number with global or local network prefix phone-context. However, VoIP Platform does not perform any translations on the To header Tel URI. Therefore, the To header SIP URI may contain anything.
Contact	SIP URI	User	Anything as allowed per the specification.
		Host	Access device domain name or IP address.
Remote Party ID, P-Asserted-Identity, P-Preferred Identity		display-name	Calling party name, if available.
	SIP URI	User	Access device user name/address of record/line/port.
		Host	Access device domain name or IP address or Application Server cluster domain name, Subscriber domain name, Application Server alias, or Application Server IP address (for devices which register).

3.3 URLs for Telephone Calls (RFC 2806)/tel URI for Telephone Numbers (RFC 3966)

VoIP Platform fully supports the “tel” URI scheme, as specified in RFC 2806. An access device may use either a “tel” URI or a SIP-URI in requests sent to VoIP Platform. SIP-URI is the recommended format for access devices to use when sending messages to VoIP Platform.

3.4 VoIP Platform will only support the revised RFC 3966 in subsequent releases.

3.5 Privacy Mechanism for the Session Initiation Protocol (SIP)/Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks (RFC 3323/RFC 3325)

VoIP Platform supports the standards track RFC 3323, A Privacy Mechanism for the Session Initiation Protocol (SIP), and RFC 3325, Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks, to protect the identity of VoIP Platform users when inter-working with network devices. Note that VoIP Platform also supports other privacy drafts. VoIP Platform does not make use of any of the privacy specifications for messages sent from VoIP Platform on the access interface. The Access Interface is treated as untrusted and all devices on the Access Interface are not considered part of the trusted network. VoIP Platform encrypts the FROM header as appropriate when sending messages on the access interface.

RFC 3325 adds two new SIP headers – P-Asserted-Identity and P-Preferred-Identity. RFC 3325 specifies that a party in a call dialog can be assigned one or more explicit identities within a trust domain indicated by one or more P-Asserted-Identity headers. These headers supersede any other headers when it comes to matching a user.

VoIP Platform supports only one identity for a given user in a SIP message. This identity includes a user name and may also include a display name. According to RFC 3325, multiple P-Asserted-Identity headers are allowed within a SIP message. However, VoIP Platform only acts upon the first one encountered in any SIP message that matches a VoIP Platform originator. Note that if one or more P-Preferred-Identity headers are present, VoIP Platform uses the first P-Preferred-Identity header encountered to attempt to identify the VoIP Platform subscriber. VoIP Platform supports only one identity for a given user in a SIP message. This identity includes a user name and may also include a display name.

VoIP Platform acts as a privacy service as described in RFC 3323. RFC 3323 adds a new SIP header called Privacy. It can be assigned one or more of the following values: “none”, “header”, “session”, “user”, and “critical”. “Header”, “session”, and “user” indicate specific areas within SIP messages where privacy should be enforced. Similarly, RFC 3325 adds “id” as a new area of privacy, and RFC 4244 adds "history". The values “none” and “critical” are defined in RFC 3323. If neither “none” nor “critical” are present, then the device is instructed to give its best effort to achieve privacy.

VoIP Platform supports the values of the Privacy header as follows:

“None”: This indicates that the privacy service must not add privacy. In terms of VoIP Platform, this means that CLID blocking should be disabled and the identity of the originator should be made available. VoIP Platform honors this value on the network interface. However, on the access interface, VoIP Platform does not honor this value and uses the setting of the VoIP Platform subscriber CLID Delivery Blocking service to determine whether the identity should be presented.

“Critical”: This indicates that the message should be rejected if the privacy service cannot or will not enforce the specified privacy. If one of the values of the Privacy header is “critical” and VoIP Platform chooses not to honor that privacy request, VoIP Platform rejects the invitation with a SIP 500 Server Error response as specified in RFC 3323. Note that this choice of return value is specified in RFC 3323 with a strength of “MUST”.

“User”: This indicates that the privacy service should enforce user-level privacy for the subscriber, by removing any user identification from the SIP message. VoIP Platform treats the privacy request as requesting “Anonymous Presentation”. There is no mechanism in RFC 3323 to specify only one of “Anonymous Name” or “Anonymous URI” as there was in previous drafts.

VoIP Platform handles “Anonymous Presentation” by replacing the value of the NameAddr in the FROM header with the following:

"Anonymous" <sip:anonymous@anonymous.invalid>

Note that this is different from the NameAddr format used in previous releases to hide the identity if the proprietary privacy support was configured:

<sip:Private@localhost>

VoIP Platform sends the new NameAddr value for all scenarios, regardless of whether or not this new version of privacy is currently configured for use in VoIP Platform.

“Header”

− “Header” indicates that the privacy service should enforce privacy for all of the headers in the SIP message, which may identify information about the subscriber. This includes the Via and Contact headers. VoIP Platform provides this capability by default as a back-to-back User Agent.
− VoIP Platform should treat the privacy request as requesting “Anonymous Presentation” detailed above under the “User” value of the Privacy header.
“Session”
− “Session” indicates that the information held in the session description (SDP) should be hidden outside of the trust domain. To accomplish this, a privacy service would have to terminate the session on one end and originate one on the other end.
− If VoIP Platform receives a request that includes a privacy header that indicates both “session” and “critical”, VoIP Platform rejects the message with a 500 Server Error response.
“Id”
− The Asserted Identity RFC 3325 adds the value “Id” to the privacy header. In such a case, the P-Asserted-Identity header is removed upon leaving the Trust Domain. In other words, if a message from the PSTN contains a P-Asserted-Identity header and also a Privacy header that contains “id”, then the P-Asserted-Identity header is not populated when the invitation is sent to an access device.
“History”
− The History-Info RFC 4244 adds the value “history” to the Privacy header. This requests that privacy be applied to the History-Info header or to specific entries when sending the header outside the trusted domain. For more information on History privacy, see section 3.7 History-Info SIP Header (RFC 4244).

As stated above, VoIP Platform never populates an outgoing message with a P-Preferred-Identity or P-Asserted-Identity header, or otherwise make use of these RFCs on the access interface. However, VoIP Platform accepts messages on the access interface from devices which include the P-Preferred-Identity or P-Asserted-Identity. VoIP Platform uses the contents of either of these headers to identify the device of the subscriber on VoIP Platform. If an association cannot be found, VoIP Platform uses the FROM header to find the device. VoIP Platform does not honor privacy requests using RFC 3323/3325 on the access interface since the interface is considered untrusted.

3.6 SIP Extensions for Caller Identity and Privacy (draft-ietf-sip-privacy-03, draft-ietf-sip-privacy-00)

VoIP Platform also supports the SIP Extensions for Caller Identity and Privacy draft to protect the identity of VoIP Platform users when inter-working with network devices. Note that VoIP Platform supports both version 00 and 03 of this draft. VoIP Platform does not use either of these drafts when sending messages on the access interface. Since the access interface is considered untrusted, VoIP Platform only populates the FROM header and does not include the Remote-Party-ID headers.
VoIP Platform accepts messages on the access interface with Remote-Party-ID headers. The Remote-Party-ID header, when present, takes precedence over the FROM header and VoIP Platform uses the Remote-Party-ID header to identify the VoIP Platform subscriber when present.
As stated in the previous section, when calling party information is restricted, VoIP Platform will send a FROM header in the INVITE with a Name-Addr as shown below:
"Anonymous" <sip:anonymous@anonymous.invalid>
3.6 Diversion Indication in SIP (RFC 5806)
VoIP Platform fully supports this functionality. Note that VoIP Platform transforms Diversion entries with a presentation indicator of restricted calling line identity to a Name-Addr of "Anonymous" <sip:anonymous@anonymous.invalid>. This transformation is done to protect the identity of the Diversion entry on the untrusted access interface.
On the access side, VoIP Platform converts the proprietary diversion-reasons used on the network side to unknown.
VoIP Platform uses the proprietary parameters to handle certain service interactions. The following parameters may be added to the Diversion header on the access side:
diversion-inhibited: This extension indicates that this call should not be diverted (forwarded). It is explained in more detail in the text that follows.
family: This extension indicates that the diversion entry was added by the Simultaneous Ringing Family service.
answered-count: This extension is internal to VoIP Platform and should be ignored.
network-inhibited: This extension is internal to VoIP Platform and should be ignored.

NOTE: The number of call redirections is obtained from the diversion-counter parameter of each diversion entry according to section 10.2.4 SIP to ISUP Translation, of RFC 5806.

3.7 History-Info SIP Header (RFC 4244)

3.7.1 Operation

VoIP Platform supports the History-Info header as defined in RFC 4244. The History-Info provides a functionality that overlaps the Diversion header described in section 3.6 Diversion Indication in SIP (RFC 5806).

3.7.2 Syntax

The History-Info SIP header format originates from the RFC 4244 [49]. VoIP Platform extensions are the same as they are for the Diversion header and are described in section 3.6 Diversion Indication in SIP (RFC 5806).

3.7.3 Interactions with Diversion Header

VoIP Platform supports incoming redirection call information through the Diversion SIP header and through the History-Info SIP header. VoIP Platform always prioritizes the History-Info header. Therefore, VoIP Platform processes only the History-Info header and ignores the Diversion header if both headers are presented within the same incoming message. VoIP Platform can be configured to generate History-Info or Diversion in outgoing messages. VoIP Platform generates the configured header independently from the type of header it received. VoIP Platform automatically converts History-Info into Diversion and Diversion into History-Info when required.
When converting, the order of the entries is reversed. The last (that is, most recent) redirection is the first entry of the Diversion header and the second-to-last entry of the History-Info header (the last entry of History-Info header is always the request URI).
When converting from History-Info to Diversion, the Index, Cause, and Text are lost.
The History-Info privacy attribute is converted by setting the privacy attribute to “full” for the specific Diversion entry. The History-Info privacy header (history, session, and header) is converted by setting the privacy attribute to “full” for all current entries (all entries present before the conversion). The Diversion privacy attribute (set to “full”, “name”, “uri”) is converted by setting the privacy attribute to the specific History-Info entry.
VoIP Platform does not transmit the History-Info index within the Diversion header. Therefore, if the system is configured to transmit the redirected call information through the Diversion header on the network side and a call goes to the network, the History-Info index is lost and regenerated.

3.8 Priority Alerting and Ring Splash

VoIP Platform supports priority alerting/distinctive ringing to access devices. Priority alerting/distinctive ringing allows a user to provision call criteria to provide a distinctive ring or call waiting alert tone when a call is received that meets the criteria. The criteria may include information such as the calling line identification, time of day, day of week, and so on.

VoIP Platform provides this service by making use of the Alert-Info header as defined in RFC 3261, section 20.4 in the INVITE message. However, instead of providing a URL for the ringing .WAV file to play in the network, VoIP Platform provides a URL specifying the access device and a distinctive ringing indicator. For example, the Alert-Info header as specified in RFC 3261 might look like the following example:
Alert-Info: <http://www.example.com/sounds/moo.wav>
An Alert-Info sent by VoIP Platform would look like the following example:
Alert-Info: <http://127.0.0.1/Bellcore-dr2>
A device receiving a URL referencing itself via the loopback IP address, 127.0.0.1, with the Bellcore keyword should play the appropriate ringing pattern as stated below.

3.9 Priority Ringing on Device and Ring Splash

VoIP Platform provides three priority alerting cadences and one ring splash cadence as specified in the LSSGR, GR-506-CORE, section 14. VoIP Platform does not send an Alert-Info header when the standard ringing pattern should be used. VoIP Platform will send down the following Alert-Info headers for the various distinctive ringing patterns:

Alert-Info: <http://127.0.0.1/Bellcore-dr2>
Alert-Info: <http:// 127.0.0.1/Bellcore-dr3>
Alert-Info: <http:// 127.0.0.1/Bellcore-dr4>
Alert-Info: <http:// 127.0.0.1/Bellcore-dr5>

For analog ground-start and loop-start lines, the ringing patterns should adhere to the requirements in GR-506-CORE. The following table maps the URLs sent by VoIP Platform to the five ringing patterns specified in GR-506-CORE section 14.

3.10 Priority Call Waiting Tone on Device

GR-506-CORE also specifies requirements for call waiting tones. For SIP phones and other intelligent devices which provide their own call control, VoIP Platform sends Alert-Info headers in INVITEs to devices for subsequent call waiting calls, with priority alerting as specified above in section 3.8.1 Priority Ringing on Device. Upon encountering an additional call, the device should adhere to the cadences, as specified in GR-506-CORE, section 14.
For other devices which rely on VoIP Platform call control, VoIP Platform sends an INFO with the following keyword in the INFO body, as specified in section 3.13 SIP INFO Method (RFC 2976).
CallWaitingTone1
CallWaitingTone2
CallWaitingTone3
CallWaitingTone4

For analog ground-start and loop-start lines, the call waiting tone patterns should adhere to the requirements in GR-506-CORE. The following table maps the URLs sent by VoIP Platform to the four call waiting tone patterns specified in GR-506-CORE section 14.

3.11 Offer/Answer and Early Media Support (The SIP UPDATE Method (RFC 3311)/Reliability of Provisional Responses in SIP (RFC 3262)/Early Session Disposition Type for SIP (RFC 3959)/Early Media and Ringing Tone Generation in SIP (RFC 3960)

VoIP Platform fully supports the offer/answer exchanges defined in RFC 3264 and provides extensive support for early media session negotiation. VoIP Platform also provides protocol enablers to support backwards compatibility with devices which are not compliant to the early media offer/answer exchanges.

3.11.1 Support for Media Changes between Early and Established Sessions

VoIP Platform provides Call Redirection services such as Call Forwarding No Answer and Voice Mail No Answer, which redirect the originating endpoint to a new device using the original offer from the originating endpoint. VoIP Platform also supports interoperating with a downstream forking proxy on the terminating side, which could lead to VoIP Platform receiving multiple 18x responses for different dialogs. To support these situations, VoIP Platform allows multiple early dialogs.
VoIP Platform behavior is configurable through system parameters at the CLI. When the accessSupportForkingProxy parameter is enabled, VoIP Platform allows the construction of multiple early dialogs, each with a different To header tag, when the originator is an access device. Similarly, when the networkSupportForkingProxy parameter is enabled, header tag, when the originator is a network gateway of softswitch.
When support for forking proxy is disabled (accessSupportForkingProxy or networkSupportForkingProxy), VoIP Platform creates a single dialog with the originator, that is, it always uses the same To header tag. If the terminating side receives responses to multiple dialogs, they get mapped to the same dialog on the originating side.
VoIP Platform can also provide error correction to allow interoperability with devices not fully compliant with RFC 3261. This option is controlled by the errorCorrectionOnSDPChange parameter.
If errorCorrectionOnSDPChange is disabled, and multiple 18x responses are received for a same dialog (same To tag) but with different SDPs, the generated 18x responses by the Application Server do not correct this violation; in this case, the generated 18x responses also refer to the same dialog on the originating end and the Application Server remembers the SDP from the last received 18x response.

If errorCorrectionOnSDPChange is enabled, the Application Server "corrects" this violation by generating 18x responses that carry a different To tag on the originating end. This option is useful to interoperate with devices that incorrectly modify their SDP answer within a dialog.

NOTE: Error correction on SDP change applies only if support for forking proxy is enabled. If support for forking proxy is disabled for the originating endpoint, then no error correction on SDP change is made, and the option has no effect.

3.11.2 2xx Media Handling

VoIP Platform provides flexible handling of 2xx responses to allow for support of devices which are not compliant to RFC 3261 and early media offer/answer exchanges. According to RFC 3261, an endpoint may send a 2xx response with no SDP if an answer SDP was provided in a reliable provisional response. The endpoint may also re-send the answer SDP in the 2xx response as long as it is the exact same SDP (same "o=" line).
VoIP Platform supports receiving a 2xx response with no SDP. For backward compatibility and to minimize interoperability impacts, VoIP Platform allows the sending of an SDP in the

2xx response to the originating endpoint even when the terminating endpoint does not include an SDP. This option is controlled by the forceAnswerSDPonAnswer parameter. If enabled, the 2xx response includes the SDP of the dialog for which the 2xx response is received when it is an answer SDP (the previously received 18x response for that dialog included an answer SDP). However, if an SDP offer/answer exchanged is initiated for that dialog with the PRACK and/or UPDATE before the 2xx response is received, the SDP is not included in the 2xx generated by the Application Server. This option is useful to interoperate with devices that expect a 2xx response to always contain an SDP.

3.11.3 Session Initiation Protocol (SIP) UPDATE Method (RFC 3311)

VoIP Platform fully supports this functionality.
NOTE: RFC 3311 section 5.1 recommends sending a re-INVITE instead of an UPDATE for confirmed dialogs.
Devices should not send the UPDATE method with SDP on confirmed dialogs. However, VoIP Platform can receive and process received UPDATE requests with SDP on confirmed dialogs. Also, if VoIP Platform receives an UPDATE request with SDP for a confirmed dialog, VoIP Platform will forward the request as-is to the other party as long as the other party advertised support of the UPDATE method. If the other party did not advertise support of the UPDATE method, then VoIP Platform will change the UPDATE request into a re-INVITE.
The UPDATE method may be used to update the SDP of a dialog or the dialog itself. The following are some of the SIP headers that the VoIP Platform Application Server allows to be updated by a received UPDATE request: Allow, Contact, Min-SE, Session-Expires, and Supported.

VoIP Platform assumes that UPDATE method is not supported within a dialog until explicitly informed otherwise by a received Allow header.
The UPDATE method is also sent by VoIP Platform for a session-timer refresh or a VoIP Platform session audit refresh if the dialog indicates that the remote party supports the UDPATE method.
When used for offer/answer exchanges, a SIP endpoint can include an offer in the UPDATE request only if it has received an answer in a reliable provisional response.
If the originating endpoint does not receive an answer in the reliable 18x response and sends an offer in the UPDATE, then VoIP Platform responds with a 400 Bad Request.
If the originating endpoint receives an answer in a non-reliable 18x response and sends an offer in the UPDATE, then VoIP Platform responds with a 400 Bad Request.

In addition, in the cases where an Access Device rejects an offer with a 488 response and includes a Warning header and/or a session descriptor, VoIP Platform only proxies the Warning header back to the endpoint that generated the offer. The session description included in the 488 response is ignored by VoIP Platform.

3.11.4 Reliability of Provisional Responses in SIP (RFC 3262)

VoIP Platform supports this functionality with some deviations from the specification for interoperability. It is strongly recommended that devices using the access interface on VoIP Platform support this functionality.
VoIP Platform handles reliable responses end-to-end, as opposed to leg-by-leg. If the 100rel configuration option is enabled on VoIP Platform, then the 100rel option tag is proxied from the originating endpoint to the terminating endpoint and "reliability of provisional responses" is negotiated between the two connecting endpoints.

VoIP Platform handles early media updates triggered with a PRACK request according to RFC 3262.
If the 100rel configuration option is enabled, then reliable responses are enabled for all calls, regardless of the transport type: udp or tcp.

Deviations on provisional response handling:
When the VoIP Platform Application Server receives a PRACK message that does not match any unacknowledged provisional response, it sends back a 200 OK instead of a 481 response as requested by RFC 3262.
When the Application Server is expecting a PRACK for a given dialog, if no PRACK is received for the provisional response the Application Server retransmits it for 64*T1 seconds and then sends back a 408 Request Timeout response. This is different from the RFC 3262 recommendation, which is to “reject the original request with a 5XX response”.

3.11.5 Early Session Disposition Type for Session Initiation Protocol (SIP) (RFC 3959)/Early Media and Ringing Tone Generation in Session Initiation Protocol (SIP) (RFC 3960)

VoIP Platform fully supports this functionality. It is strongly recommended that devices using the access interface on VoIP Platform support this functionality.
RFC 3959/3960 defines a method for providing early media in an independent session. The terminating endpoint provides an early-session offer in the 18x response, and the originating endpoint provides the early-session answer in the PRACK. As such, beyond supporting early sessions, the two endpoints must also support reliable responses.
Endpoints that support or require early sessions must include the early-session option tag in the Supported or Require header of the INVITE request.
VoIP Platform enables early session negotiation between two endpoints by proxying the early-session option tag from the originating endpoint to the terminating endpoint. In addition, if the terminating endpoint provides an early offer SDP, then this SDP is also proxied back to the originating endpoint. The following call flows show how the early-session option tag and the early-session SDPs are proxied across VoIP Platform.

If a call topology change occurs and the originating dialog is still in the alerting state, then the early-session option tag is proxied to the new terminating endpoint such that an early-session can also be negotiated between the two endpoints. When this happens, the originating endpoint receives an 18x response with a new early-session offer and a different To-tag. The originating endpoint can then respond with a new early-session answer
If a call topology change occurs and the originating dialog is active (confirmed), then the early-session option tag is not proxied to the new terminating endpoint.

3.12 Session Timers in Session Initiation Protocol (SIP) (RFC 4028)

VoIP Platform fully supports this functionality. VoIP Platform will use the UPDATE method to refresh a session, if the remote user agent indicates that it supports the UPDATE method via the ALLOW header. If UPDATE is not supported (or indicated) by the remote user agent, VoIP Platform uses a re-INVITE to refresh the session. According to section 7.4 in RFC 4028, to determine if a session is still active, VoIP Platform checks the origin line of the SDP of a re-INVITE to determine if the SDP has changed. If the origin line has not changed, VoIP Platform treats the re-INVITE like a session extending/auditing re-INVITE.
VoIP Platform does not require the session timer functionality because VoIP Platform has its own session audit capability. However, both session timer and the VoIP Platform session audit can be used simultaneously.
With the addition of this support, a new parameter has been added to enable/disable Session Timer support. The new parameter is sessionTimer. The default value is “false”. When this parameter is “true”, VoIP Platform advertises support of the Session Timer functionality. When this parameter is “false”, VoIP Platform does not advertise support of Session Timer.
By default, VoIP Platform never includes the Session Expires header in requests including session refresh requests. VoIP Platform only includes the session-expires header within 200 responses when the sessionTimer parameter is enabled and the request includes the session-expires header. VoIP Platform follows RFC 4028 to determine the refresher for the session; however, the remote user agent is always preferred by VoIP Platform and chosen when possible.
VoIP Platform can also be configured to explicitly request the SIP session timer when the device (or a proxy) supports it. This means that if the Application Server receives an INVITE with the timer option in the Supported header, but no Session-Expires header, it sends the timer option in the Require header and the Session-Expires header in the 200 OK response. In addition, when the Application Server sends an INVITE request, it includes the Session-Expires header

directly. If the target device or an intervening proxy does not support this option, the 200 OK response simply does not contain the Session-Expires header.

In both of these scenarios, the outgoing Session-Expires header contains the configured preferred session timer value (from AS_CLI/System/CallP/SessionAudit>sipSessionExpiresTimer) and sets the refresher to the configured value (the new configurable parameter is AS_CLI/System/CallP/SessionAudit> preferredSessionTimerRefresher). When the preferredSessionTimerRefresher is set to “local”, the Application Server sets the refresher parameter so that it controls the refreshes (that is, “uas” in 200 OK responses and “uac” in INVITE requests). When this parameter is set to “remote”, the Application Server tries to get the far end to handle the refreshes (by setting the refresher parameter to “uac” in 200 OK responses and “uas” in INVITE requests).

Additionally, the minimum allowed value for the sessionExpiresMinimum is changed from “0” to “30”.

3.13 SIP INFO Method (RFC 2976)

VoIP Platform supports this functionality. When the INFO message contains a message body configured with a configured Content-Type, VoIP Platform proxies the INFO message to the remote party.

For example, applications that support video codecs such as H.263 need to convey media control information between participating devices. While it is possible to convey such information directly in the media streams themselves, certain information is considered closely related to the application logic. This higher-level application information is best conveyed through the signaling channel, rather than through the media stream. If the applications use SIP signaling, such information is conveyed in SIP INFO requests.

For more details on message body proxy capabilities, see section 3.21.1 VoIP Platform Content-Type Support.

In addition to proxying INFO messages, VoIP Platform processes some specific INFO content. VoIP Platform uses the INFO method to support flash-based services on access devices such as SIP gateways, which provide a SIP interface for analog (FXS) lines. VoIP Platform also recognizes DTMF signals conveyed by INFO messages in the following formats: application/dtmf-relay, application/dtmf, and audio/telephone-event.

3.13.1 Flash-based Service Support via INFO Method

VoIP Platform uses a proprietary extension to the INFO method to support flash-based user services. A device must support this extension for VoIP Platform to provide flash-based services such as call waiting, call transfer, three-way calling, and so on. Specifically, the extension includes the definition of a new value for the Content-Type header. The new value is: application/broadsoft.

The application/broadsoft Content-Type allows an endpoint to notify VoIP Platform that a flash hook has occurred or to direct an endpoint to play a tone, as specified by the VoIP Platform Application Server.

The Content-Type of application/broadsoft indicates that a proprietary body is in the message. The body must be in one of the following formats for VoIP Platform or the endpoint to interpret the intention: (These fields are not case sensitive.)

event <event name>
play tone <tone name>
stop <tone name>

Optionally, the play tone body may contain the following body parts to communicate call waiting calling party identification information. Note that the INFO body is case insensitive.

Calling-Name:<calling-name> where <calling-name> is a string representing the calling party’s name
Calling-Number:<calling-number> where <calling-number> is a string representing the calling party’s number

The Calling-Name and Calling-Number are always included in the INFO for call waiting as long as the calling party information is available. When the information is not available, the device must populate the calling line identification signal to the analog line with the appropriate unavailable signal. When the calling party information is not available, the Calling-Name and/or Calling-Number fields are not included in the INFO method body. It is possible that the calling number may be available without the calling name and vice versa. When these conditions occur, only the information that is available is included in the INFO method body (that is, it is possible to have a Calling-Number field in the INFO method body without a Calling-Name field and vice-versa).

When any portion of the calling party information is restricted, the Calling-Name and Calling-Number fields are included in the INFO method body header and are populated with “Private”.

NOTE: Restricted calling party information overrides unavailable calling party information.

When the calling number is restricted and the calling name is unavailable or vice versa, both the Calling-Name and Calling-Number fields are included in the INFO method body and are populated with “Private”.

3.13.2 Video Support via INFO Method

VoIP Platform supports the ability to proxy INFO requests after a call has been established that convey media control information for video calls. The INFO requests convey the media control information in the request body, which has the MIME type application/media_control+xml.
The Application Server proxies the request with end-to-end reliability. In other words, the Application Server sends back a 200 response to the INFO request after it receives a 200 response from the forwarded INFO request.
The specific media control information that the INFO request conveys includes:
Video Picture Fast Update Request (decoder to encoder)
Video GOB Fast Update Request (decoder to encoder)
Video MB Fast Update Request (decoder to encoder)
Video Picture Freeze Request (encoder to decoder)

The H.263 standard provides more information about these requests. The Internet draft entitled “XML Schema for Media Control” describes the encoding of this media control information in an XML payload with MIME type application/media_control+xml. For more information, see XML Schema for Media Control [37].
The following figure shows one possible scenario where the Application Server must proxy SIP INFO messages between devices. Audio and video streams are established between a video phone and a gateway. Signaling is handled by the Application Server via SIP messages. When the gateway needs to convey media control information to the video phone, it sends a SIP INFO message to the Application Server, which forwards it to the video phone. Similarly, when the video phone needs to convey media control information to the gateway, it sends a SIP INFO request to the Application Server, which forwards it to the gateway.

3.14 Session Initiation Protocol (SIP) Refer Method (RFC 3515)/SIP “Replaces” Header (RFC 3891)/SIP Referred-BY Mechanism (RFC 3892)

VoIP Platform supports this functionality.
VoIP Platform never sends a REFER method or any method with a Replaces header. However, VoIP Platform accepts receiving a REFER method for blind transfer and a REFER method with Replaces header overloaded into Refer-to header for transfer with consultation.
In general, VoIP Platform does not support receiving a Replaces header in an INVITE. VoIP Platform ignores the Replaces header field and processes the INVITE as if the Replaces header is not present, regardless of the presence of the Require header. However, VoIP Platform supports INVITE messages with a Replaces header in some

Shared Call Appearance scenarios. For information on the use of the Replaces header for Shared Call Appearance, see VoIP Platform SIP Access Side Extensions Interface Specification [43].
Because VoIP Platform is a back-to-back user agent (B2BUA), VoIP Platform has knowledge of all calls to/from a particular device. As such, VoIP Platform is able to accept a REFER method and perform the appropriate call control requests within VoIP Platform, rather than proxy the request on to the other party in the call. This is vital for interworking SIP devices with devices which support other protocols such as MGCP. This is also beneficial for interworking with SIP devices which do not support the REFER method.
Access devices may use this method and header to trigger transfer services within VoIP Platform. VoIP Platform will transparently process the request such that the access device is unaware of any interworking required to complete the request.
Depending on configuration, VoIP Platform can send or suppress the NOTIFY for the implicit subscription created by the REFER method on the dialog. When configured to do so, the Application Server sends a NOTIFY request per RFC 3515 to suppress the implicit REFER subscription. The NOTIFY terminates the implicit subscription and contains message/sipfrag content as described in RFC 3515. When configured not to send the NOTIFY, the BYE implicitly terminates the subscription.

3.15 Session Initiation Protocol (SIP) Call Control Conferencing for User Agents (RFC 4579)/Session Initiation Protocol (SIP) Event Package for Conference State (RFC 4575)/Framework for Conferencing with SIP (RFC 4353)

VoIP Platform supports the ad-hoc conferencing methods defined in section 5.4 of RFC 4579. A SIP device establishes a dialog with a conference bridge. Then the device refers existing dialogs to the conference bridge that automatically mixes the parties as they are added to the bridge. VoIP Platform does not support RFC 4575. However, VoIP Platform will support this RFC in future releases to allow enhanced network conference call control for the devices.
When a conference is initiated, VoIP Platform generates a Conference-Id distinct from the Conference URI. The Conference-Id is a SIP URI constructed by the Application Server and provided in the Contact header of the response to the conference creation INVITE request. The Conference-Id is built based on the following:
The user part of the SIP URI is the user part of the Conference URI. Note that the Conference URI is configured at the service provider level as either the system default or a service provider-specific SIP URI.
The host and port parts of the SIP URI are constructed using the Application Server access-side address and SIP-listening port.

The Conference-Id is a globally routable URI that enables proper routing of the subsequent ACK and REFER messages. In particular, it allows multiple Application Server clusters to share the same Conference URI. When it receives the SIP REFER messages, the Application Server applies the following rules to the Refer-To header:
If the Refer-To header matches the Conference-Id, the associated dialog is transferred to the conference. The Refer-To header matches the Conference-Id if they have the same user and host parts. Ports and parameters are ignored.
If the Refer-To does not match the Conference-Id, but does match the Conference URI, then the transfer is rejected.
If the Refer-To matches neither the Conference-Id nor the Conference URI, then the transfer request is ignored by the device-initiated conference logic. It may be processed by other VoIP Platform services however, such as Call Transfer.

A sample flow is shown for a SIP access device initiating network-based conferencing, using the ad-hoc method defined in section 5.4 of RFC 4579. Note that depending on configuration, VoIP Platform' Application Server can send or suppress the NOTIFY message following the REFER request. The following flow suppresses the NOTIFY, see section 3.14 Session Initiation Protocol (SIP) Refer Method (RFC 3515)/SIP “Replaces” Header (RFC 3891)/SIP Referred-BY Mechanism (RFC 3892) for an example NOTIFY message.

3.16 SIP-specific Event Notification (RFC 3265)

VoIP Platform fully supports this functionality. This functionality provides VoIP Platform with a powerful service creation platform that can be extended by adding support for additional event packages. In Release 9.0, VoIP Platform supports the following event packages on the access interface:
Call-info (VoIP Platform proprietary event package for support of enhanced shared call appearances for key system emulation and enhanced business applications. For more information, refer to VoIP Platform SIP Access Side Extensions Interface [43].
Line-Seize (VoIP Platform proprietary event package for support of enhanced shared call appearances for key system emulation and enhanced business applications). For more information, refer to VoIP Platform SIP Access Side Extensions Interface [43].
Message-summary (message-summary). For more information, refer to A Message Summary and Message Waiting Indication Event Package for the Session Initiation Protocol (SIP) [17].
Microsoft Windows Messenger version 4.7 presence support:
− VoIP Platform will proxy SUBSCRIBEs and facilitate the peer-to-peer presence requests generated by Microsoft Windows Messenger and other compatible devices.
− These SUBSCRIBEs are not compliant to RFC 3265 since they do not contain an Event header and they have no package name.

Support for additional event packages will be added to VoIP Platform in the future, enabling a wide variety of service solutions.
VoIP Platform currently does not support the ability to share a dialog with calls and subscriptions. Additionally, VoIP Platform does not support multiple subscriptions on a dialog at the same time.

3.17 RTP Payload for DTMF Digits (RFC 4733)

VoIP Platform supports this specification. The VoIP Platform Media Server uses RFC 4733 to provide the ability to collect DTMF digits for IVR services. If the device connecting to the Media Server supports RFC 4733 as indicated in the SDP, the VoIP Platform Media Server uses RFC 4733 to collect DTMF digits from the device. Otherwise, the VoIP Platform Media Server collects the digits via the RTP stream of packets.
Access devices should support RFC 4733 as it guarantees reliable delivery of DTMF digits from the access device to the entity collecting the digits.

3.18 VoIP Platform Content-Type Support

VoIP Platform supports session establishment for requests with application/sdp content types. In addition, VoIP Platform also supports session establishment of other Content-Types via system configuration. The application/sdp is the only content-type allowed by the Application Server by default. The VoIP Platform Application Server rejects unrecognized content-types not explicitly provisioned on the Application Server with a 415 Unsupported Media Type response.
Requests with bodies with multipart/mixed types are also implicitly supported, requiring no configuration. The following rules summarize VoIP Platform' content-type handling in multipart/mixed bodies:
All content-types recognized by VoIP Platform (that is, explicitly configured) are allowed and proxied by VoIP Platform.
VoIP Platform maintains the multipart/mixed body when more than one content-type is recognized by VoIP Platform.
All content-types not recognized by VoIP Platform are silently discarded when contained within a multipart/mixed body.
If no content-types are recognized by VoIP Platform in a multipart/mixed body, VoIP Platform rejects the request with a “415 Unsupported Media Type” response.
If only one content-type is recognized by VoIP Platform, the multipart/mixed body is dropped and replaced with a normal body containing the single content-type.

All content-types added to the supported content-type list via the Application Server CLI are proxied by VoIP Platform when received. VoIP Platform performs content sensitive processing to the following content-types: application/sdp, application/gtd, application/broadsoft, application/dtmf-relay, audio/telephone-event, and application/dtmf. VoIP Platform does not perform any context-sensitive processing on other content-types added to the supported content-type list via the Application Server CLI; these content-types are simply proxied to the remote party.
VoIP Platform does not currently use the Content-Disposition header for Content-Type header processing.

3.19 RTP: Transport Protocol for Real-Time Applications (RFC3550)/ RTP: Transport Protocol for Real-Time Applications (RFC 1889)/RTP Profile for Audio and Video Conferences with Minimal Control (RFC 3551)/RTP Profile for Audio and Video Conferences with Minimal Control (RFC 1890)

VoIP Platform uses this functionality within the Media Server to provide media resources for voice mail, IVR, conferencing, and so on.
Note that the VoIP Platform Media Server supports the following encodings:
G.711 u-law
G.711 a-law
G.726-32
G.729a

Access devices must support codec renegotiation via SIP when using codecs other than G.711 u-law, G.711 a-law, G.726-32, or G.729a to interface with the VoIP Platform Media Server. For calls using resources on a Media Server, VoIP Platform must renegotiate the media stream to G.711 u-law, G.711 a-law, G.726-32, or G.729a.

3.20 Connected Line Identification Presentation (COLP)

The Connected Line Identification Presentation (COLP) service provides the calling party with the ability to be presented with the identity of the connected party, which may or may not be the dialed party.

3.20.1 VoIP Platform Sending COLP

When Connected Line Identification Presentation (COLP) is provided by VoIP Platform, it is sent in 18x and 200 OK responses to the initial INVITE requests, as well as UPDATE and re-INVITE requests.
The header used to provide COLP depends on the setting of the privacyVersion SIP system parameter. Note that IP Multimedia Subsystem (VoIP Platform) mode and distributed group call (DGC) signaling always function as if the privacyVersion system parameter is set to "RFC 3323".
If the privacyVersion system parameter is set to "RFC 3323", then COLP is provided via the P-Asserted-Identity header.
If the privacyVersion system parameter is set to "privacy-03" or "privacy-00", then COLP is provided via the Remote-Party-ID header.
If the privacyVersion system parameter is set to any other value, then COLP is not provided in the SIP messages.

The COLP included in the PAI/RPID header is always a SIP URI entry. If the COLP is an E.164 phone number, is being sent to a user's access device without the "E164 Capable" device option, and the phone number's country code matches the user's country code, then the phone number is normalized to the appropriate prefixed national format for the country code. Otherwise, the COLP is sent without normalization.
The current COLP for a call is only provided in the message sent to a user's device if they have the COLP service enabled for the call and the privacyVersion system parameter setting allows for COLP to be included. If they do not have the COLP service enabled for the call, then the PAI/RPID header for COLP is not included in the responses.

3.20.2 VoIP Platform Receiving COLP

COLP received for a user is always ignored. The Application Server always uses the appropriate configured identity for its users and does not allow a user's device to override it.

3.21 P-Called-Party-ID SIP Header

The VoIP Platform Application Server supports the P-Called-Party-ID (PCPI) SIP header defined in RFC 3455 [47].
For a VoIP Platform user origination, the Application Server proxies the P-Called-Party-ID header in an initial INVITE in non-VoIP Platform deployments.
For a VoIP Platform user termination, the Application Server proxies the P-Called-Party-ID header in an initial INVITE if the destination is the user's primary location. It does not proxy the PCPI header to the user's secondary or alternate locations.

The following is an example of this header:
P-Called-Party-ID: sip:user1-business@example.com
VoIP Platform Application Server optionally inserts a new P-Called-Party-ID header in the outgoing request message. The content of the header reflects the terminating user identity that is reached. This identity can be the user's main address or one of his alternate addresses. If the Application Server has received a P-Called-Party-ID header in the incoming request, the latter is discarded and the newly built one is sent along with the outgoing termination request.

3.22 Via Header

VoIP Platform constructs the Via header according to rules specified in RFC 3261. Two cases must be considered. In the usual case, the branch parameter is constructed by appending the following components separated by "-" characters:
The prefix "z9hG4bKVoIP Platform."
An encoded hash value representing the host address
The destination IP and port separated by a V

An internal index associated with the destination
The message sequence number and the From header tag separated by a "-" or a "A" for ACK of INVITE 2xx responses.

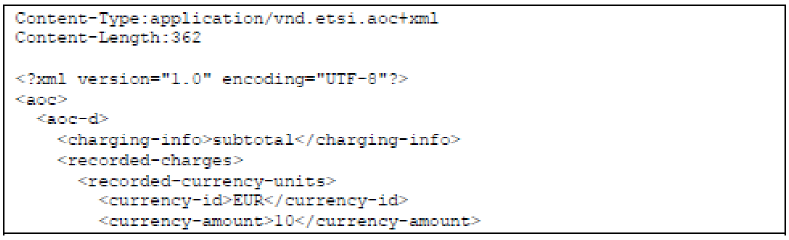
3.23 SIP Join Header

In general, VoIP Platform does not support receiving a Join header in an INVITE message. VoIP Platform rejects the INVITE message with a 481 error code.

3.24 Advice of Charge

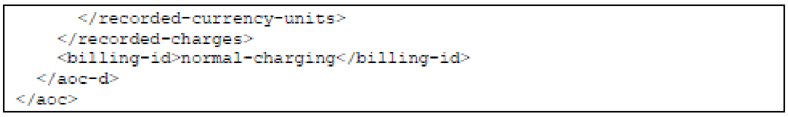
VoIP Platform can provide Advice of Charge (AoC) information to access devices, specifically AOC-D (during a call) and AOC-E (end of call).
VoIP Platform sends AoC information to access devices in message bodies encoded as application/vnd.etsi.aoc+xml.
The Application Server may send this body type in INFO messages for AoC-D and in BYE or 200 OK messages for AoC-D and AoC-E. It can also be used in a 487 Request Terminated response on a terminated call leg for both AoC-D and AoC-E.
The syntax for the application/vnd.etsi.aoc+xml is defined in 3GPP TS 24.647 v8.0.0 Advice of Charge (AoC)

Following is an example body for AoC-D:



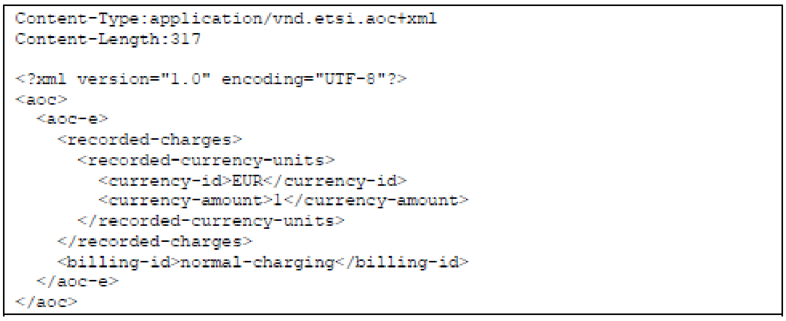
```
Content-Type:application/vnd.etsi.aoc+xml
Content-Length:362

<?xml version="1.0" encoding="UTF-8"?>
<aoc>
  <aoc-d>
    <charging-info>subtotal</charging-info>
    <recorded-charges>
      <recorded-currency-units>
        <currency-id>EUR</currency-id>
        <currency-amount>10</currency-amount>
```



```
      </recorded-currency-units>
    </recorded-charges>
    <billing-id>normal-charging</billing-id>
  </aoc-d>
</aoc>
```

Following is an example body for AoC-E:



```
Content-Type:application/vnd.etsi.aoc+xml
Content-Length:317

<?xml version="1.0" encoding="UTF-8"?>
<aoc>
  <aoc-e>
    <recorded-charges>
      <recorded-currency-units>
        <currency-id>EUR</currency-id>
        <currency-amount>1</currency-amount>
      </recorded-currency-units>
    </recorded-charges>
    <billing-id>normal-charging</billing-id>
  </aoc-e>
</aoc>
```

4 TECHNICAL SPECIFICATION OF SIP UNI VOIP END USER INTERFACE – specific interface for Fixed Network Voice Service Replacement

Specification of VoIP protocols for single users with registration to SIP server.
Specifikation is relevant for MSANs, SIP phones and for Customer premises GW (CPG) with analogue phone user interface Z and ISDN BRI interface.

4.1 List of standards applied to IP Phone, MSAN and CPGs SIP interface

3GPP TS 24.229 V8.5.0 Internet Protocol (IP) multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3
RFC 3261: SIP: Session Initiation Protocol, June 2002
RFC 3262: Reliability of Provisional Responses in SIP, June 2002
RFC 3265: Session Initiation Protocol (SIP)-Specific Event Notification, June 2002
RFC 3311: The Session Initiation Protocol (SIP) UPDATE Method, September 2002
RFC 3323: A Privacy Mechanism for the Session Initiation Protocol (SIP), November 2002
RFC 3325: Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks, November 2002
RFC 3326: The Reason Header Field for the Session Initiation Protocol (SIP)”, RFC 3326, December 2002
RFC 3455: Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the Third-Generation Partnership Project (3GPP), January 2003
RFC 3515: The Session Initiation Protocol (SIP) Refer Method, April, 2003
RFC 3455: Private Header (P–Header) Extensions to the Session Initiation Protocol (SIP) for the 3rd–Generation Partnership Project (3GPP)
RFC: 3455: Private Header (P–Header) Extensions to the Session Initiation Protocol (SIP) for the 3rd–Generation Partnership Project (3GPP),
RFC 3891: The Session Initiation Protocol (SIP) "Replaces" Header, September 2004
RFC 3892: The Session Initiation Protocol (SIP) Referred-By Mechanism, September 2004
RFC 3903: Session Initiation Protocol (SIP) Extension for Event State Publication
RFC 4028: Session Timers in the Session Initiation Protocol (SIP). RFC 4028, Internet Engineering Task Force, April 2005.
RFC 4566: SDP: Session Description Protocol RFC 4566, Internet Engineering Task Force, July 2006.
RFC 4244: An Extension to the Session Initiation Protocol (SIP) for Request History Information
RFC 5009: Private Header (P–Header) Extension to the Session Initiation Protocol (SIP) for Authorization of Early Media
Diversion Indication in SIP, draft–levy–sip–diversion–08 (expired February 2005)
RFC 3023: XML Media Types RFC 3023
RFC 2183: Communicating Presentation Information in Internet Messages: The Content-Disposition Header Field
3GPP TS 24.647 V8.1.0: 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Advice Of Charge (AOC) using IP Multimedia (IM) Core Network (CN) subsystem (Release 8)
3GPP TS 29.658 V8.4.0 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; SIP Transfer of IP Multimedia Service Tariff Information; Protocol specification.
3GPP TS 24.615 V8.2.0 (2009-06) 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Communication Waiting (CW) using IP Multimedia (IM) Core Network (CN) subsystem; Protocol specification; (Release 8);
RFC 3428: Session Initiation Protocol (SIP) Extension for Instant Messaging, December 2002
RFC: 2976: The SIP INFO Method
RFC 3420: Internet Media Type message/sipfrag;
RFC 3688: The IETF XML Registry

RFC 4575: A Session Initiation Protocol (SIP) Event Package for Conference State
TS 24.423 V8.1.0 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN);
PSTN/ISDN simulation services; Extensible Markup Language (XML) Configuration Access Protocol (XCAP) over the Ut interface for Manipulating NGN PSTN/ISDN Simulation 3GPP Services
draft-bliss-callcompletion-03: Call Completion for Session Initiation Protocol (SIP);
draft-kaplandispatch-session-id-00 A Session Identifier for the Session Initiation Protocol (SIP),
RFC 4826: Extensible Markup Language (XML) Formats for Representing Resource Lists;
RFC 5364: Extensible Markup Language (XML) Format Extension for Representing Copy Control Attributes in Resource Lists;
RFC 3863: Presence Information Data Format (PIDF)
RFC 4661: An Extensible Markup Language (XML)-Based Format for Event Notification Filtering;
RFC 4480: RPID: Rich Presence Extensions to the Presence Information Data Format (PIDF);
draft-vanelburg-sipping-private-network-indication-03 The Session Initiation Protocol (SIP) P-Private-Network-Indication Private-Header (P-Header);
RFC 5079: Rejecting Anonymous Requests in the Session Initiation Protocol;
RFC 2045: Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies;
RFC 2046: Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types;
draft-jones-sip-options-ping-02.txt (Expires: January 1, 2011) Using OPTIONS to Query for Operational Status in the Session Initiation Protocol (SIP),
3GPP TS 24.628 V8.3.0 (2009-12) 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Communication Waiting (CW) using IP Multimedia (IM) Core Network (CN) subsystem; Protocol specification; (Release 8);
RFC 5502: The SIP P-Served-User private header (P-Header) for the 3GPP IP Multimedia (IM) Core Network (CN) Subsystem
RFC 3327: Session Initiation Protocol (SIP) Extension Header Field for Registering Non-Adjacent Contacts;, Internet Engineering Task Force, 2002.
3GPP TS 24.237 V10.5.0 IP Multimedia Subsystem (IMS) Service Continuity – Stage 3;
RFC 4538: Request Authorization through Dialog Identification in the Session Initiation Protocol (SIP); 2006
RFC 6068: Session Initiation Protocol (SIP) INFO Method and Package Framework;
draft-holmberg-sipcore-proxyfeature-04, 2011: Indication of features supported by proxy:
3GPP TS 24.654 10.2.0 Closed User Group (CUG) using IP Multimedia (IM) Core Network (CN) subsystem, Protocol Specification
RFC 3680, March 2004: Session Initiation Protocol (SIP) Event Package for Registrations,
3GPP TS 24.237 v.10.5.0, IP Multimedia Subsystem (IMS) service continuity
RFC4412: Communications Resource Priority for Session Initiation Protocol

4.2 Supported SIP methods

4.2.1 SIP Requests

The following table shows the SIP methods supported by MMTEL AS. Details are included as to whether the method is only accepted and passed on by MMTEL AS, whether MMTEL AS generates the method, if the method may be used to create a new dialog and details of when the method is used.

Supported SIP Methods

Method	UAS *1)	UAS *1)	Usage
INVITE	N/E	N/E	
ACK	E	E	
CANCEL	N/E *2)	N/E *2)	

BYE	E	E	
PRACK	E	E	
UPDATE	E	E	
REFER	E	E	
NOTIFY *3)	E	N/E	NOTIFY is sent and received as part of the SUBSCRIBE dialog for Communication Completion or Conference (both ad-hoc and scheduled type) and an indication of status for a REFER request. NOTIFY is sent outside of established dialog for Dial Tone Management indication, for reporting User Location Info, and for Emergency Call start/stop indication. NOTIFY is received as part of the SUBSCRIBE dialog for SCC AS to obtain the served user's Registration Information.
REGISTER	N		
SUBSCRIBE	N/E	N/E	SUBSCRIBE is sent and received as part of Communication Completion service and sent to the Presence Server to obtain served user Presence status. SUBSCRIBE accepted by the Conference MMTEL AS (both for ad-hoc and scheduled conferences) on an existing dialog to notify participants of the status and status changes of conference participants. SUBSCRIBE request to conference events received outside a dialog or within an early dialog is rejected by the Conference MMTEL AS with "403 Forbidden" response. SUBSCRIBE is sent as part

			of the SCC AS services to obtain the served user's Registration Information from the registrar (S-CSCF).
MESSAGE		S	MESSAGE is sent as an indication to the served user by Communication Diversion service.
OPTIONS	S/E	E	May be sent outside a dialog to one of the configured addresses defined in MMTEL AS, as a ping to the MMTEL AS, which will respond without a SDP body, ref [44]. May contain a 'Supported' header. May be sent outside a dialog to an address not defined in MMTEL AS. MMTEL AS will answer with 501 Not Implemented response. Maybe sent in an early dialog to MMTEL AS which will respond with 200 OK without SDP. When MMTEL AS has one well defined session between incoming and outgoing side, MMTEL AS is transparent to OPTIONS request and response. OPTIONS may be sent in an existing conference dialog to the MMTEL AS Conference server which will respond with details of Conference server capabilities. OPTIONS maybe sent in an established dialog of a 3rd party call. MMTEL AS will respond with 200 OK with SDP. OPTIONS sent in an established incoming dialog when no corresponding outgoing, established dialog

			exists will be answered by MMTEL AS with 200 OK without SDP.
INFO	E	E	
PUBLISH	E	E	PUBLISH is sent and received as part of the Communication Completion service

Notes:
*1) UAS and UAC indicate if the method is supported when received by MMTEL AS from another node or is sent from the MMTEL AS respectively. The table indicates if the SIP method may be sent or received on; an existing dialog – E, as a new dialog creating request – N, or as a standalone transaction – S.
*2) CANCEL is only sent for an INVITE transaction but may be sent for an INVITE for which there is no existing dialog, or for a 're-INVITE' sent on an existing dialog.
*3) NOTIFY is a dialog creating response to a SUBSCRIBE Request. Where E is included for a NOTIFY Request the NOTIFY acts as the dialog creation message.

4.2.2 SIP Responses

SIP responses fall into two major categories:
- Error Responses – generated by the MMTEL AS node software to indicate errors in the received SIP message.
- Service Responses – generated by deployed services as responses to usage of specific deployed services.
The same SIP response may be used for both MMTEL AS Node and deployed service responses, e.g. "500 Internal Server Error" may be generated by the platform or from a service where an error has occurred.
Responses are defined in RFC 3261 [4] except where specifically referenced.

4.2.2.1 Error Responses

Error responses to SIP methods are summarized in Table. These responses are generated by the platform due to errors in the Request received. The warning, , is added to the error response as additional information about the node issuing the response.

Status Code	Reason Phrase	MMTEL AS could not process the request because…	Warning Code
400	Bad Request	The request could not be understood due to malformed syntax.	399
403	Forbidden	The request is not allowed outside a dialog.	399
405	Method Not Allowed	The request is not allowed.	399
408	Request Timeout	Request Timeout Other SIP servers down the signalling path did not respond in a timely fashion.	399
416	Unsupported URI scheme	The Request URI is neither a tel URI nor an embedded tel SIP URI.	399
420	Bad Extension	The "199" option-tag is	399

		included in the Require and/or Proxy-Require header field of the dialog establishment request, e.g. INVITE.	
422	Session Interval Too Small	The session expiry interval suggested in the INVITE Request was smaller than the configured minimum value (RFC4028)	399
480	Temporarily unavailable	The served user is temporarlity unavailable.	399
481	Call/Transaction Does Not Exist	The request could not be identified as part of any known session.	399
487	Request Terminated	The call was cancelled	399
488	Not Acceptable Here	The UPDATE request is received on an early dialog that is terminated.	399
491	Request Pending	MMTEL AS has another request pending.	399
500	Internal Server Error	May be received from external systems that did not perform as expected or due to internal processing error.	399
501	Not Implemented	MMTEL AS could not recognize the request method.	399
503	Service Unavailable	The service was disabled or system is overloaded.	399

Text associated with the Warning Code is dependant upon the reason the response was generated.

4.2.2.2 Service Responses

The following SIP responses may be provided by the services deployed on the MMTEL AS platform. These responses are generated as a result of normal session handling. These responses are divided into 2 categories:
- Provisional Responses, 1xx series
- Final Responses – These may be either acceptance of the SIP method, 2xx series, or rejection of the SIP method, 3xx, 4xx, 5xx and 6xx series.

4.2.2.2.1 Provisional Responses

Provisional responses are only sent or received in response to an INVITE Request.

Provisional Responses

Status Code	Reason Phrase	Comment
100	Trying	
180	Ringing	
181	Call is beeing forwarded	
182	Queued	
183	Session Progress	
199	Early Dialog Terminated	Like all the others, it may happen that 199 as the first provisional response is creating a new early dialog

4.2.2.2.2 Final Responses

The warning, is added to the error response as additional information about the node issuing the response.

Final Responses

Status Code	Reason Phrase	Comment	Warning Code
200	OK	Acceptance for all requests except REFER	
202	Accepted	Acceptance of REFER (RFC3515)and SUBSCRIBE	
301	Moved temporarily		
400	Bad Request		399
403	Forbidden		399
404	Not found		399
406	Not Acceptable		
408	Request timeout		399
410	Gone		399
416	Unsupported URI		399
420	Bad extension		
421	Extension required		
433	Anonimity Disallowed	RFC5079	399
480	Temporarily unavailable		399
486	Busy here		399
487	Request terminated		399
488	Not acceptable here		399
500	Internal server error		399
502	Bad gateway		399
503	Service unavailable		399
603	Decline		399
606	Not acceptable		399

4.2.3 SIP Headers

The MMTEL AS will transparently pass SIP headers except those listed in table below. Headers shown in table may be modified by the MMTEL AS.
Headers in received SIP requests or Responses not included in Table 8 will be ignored.
MMTEL AS will include the headers included in table where applicable.
MMTEL AS usage is only shown where the MMTEL AS specifically inserts these values into the header in a method and the usage is not specifically defined in the referenced RFC.

SIP Headers

Header	MMTEL AS Usage
Accept	MMTEL AS will pass through received values and insert content types for bodies which the MMTEL AS is prepared to receive.
Accept–Contact	MMTel specific feature tags may be added.
Alert-Info	A communication waiting Alert-info header may be added.
Allow	MMTel AS will update received contents to include SIP Methods used to support specific MMTel services.
Call ID	MMTEL AS creates new Call-Id values which are mapped between received and sent messages.
Call-Info	MMTEL AS will remove the informational elements from the Call-Info header of the SIP messages that are sent by the served user if the informational element has the purpose=calltransfer; m=consultative parameters.

Contact	A new Contact is included in a sent initial Request. MMTEL AS maps the Contact header in subsequent requests and responses between received and sent messages.
Content-Disposition	When this header is absent the default values for content disposition and handling for a body are implied.
Content-Lenght	
Content-Type	
Cseq	
Diversion	
Event	
Expires	
Feature-Caps	If SRVCC for alerting calls is enabled, SCC AS will include the header in SIP INVITE or SIP 1xx/2xx responses where applicable,
From	From header may be modified between received and sent initial requests. MMTEL AS generates a new from-tag for a sent initial request. The From header , including from-tag, is mapped between received and sent responses and subsequent requests.
History–info	Inserted to indicated call diversion. Existing entries may be modified to include served user privacy settings.
Info-Package	SRVCC for alerting calls includes Info-Package in SIP INFO
Max–Forwards	
MIME–Version	
Min–Expires	
Min–SE	
Organization	
Path	
P–Access–Network–Info	
P–Asserted–Identity	
P-Area-Info	Used by the Japanese Charging service for ICBS data.
P–Charging–Function–Addresses	
P–Charging–Vector	In addition to usage in SIP specification [10], it is used by Japanese Charging service for ICBS and FCH data.
P-Com.TimeZone	Carries Time Zone and DST info.
P-Com. PrivateUserID	Indicates the IMS Private User ID (IMPI).
P-Com.User-Equipment-Info	Carries info on user equipment.
P–Early–Media	MMTEL AS includes in provisional responses when an announcement is to be played prior to session establishment or rejection. MMTEL AS will pass through a PEarly-Media received in a provisional response
P-Private-Network-Indication	
P-Served-User	The attribute MMTEL ASSipSupportPServedUserHeader [30] defines if the P-Served-User header is supported. When supported, the served user is primarily determined from the received P-Served-User header. The P-Served-User header can be inserted by MMTEL AS for Call Out of Blue sessions
Priority	
Privacy	Inserted or modified based upon served user privacy settings or when served user privacy is applied, prior to

	sending Request or Response to user.
Rack	
Reason	Q.850 cause values in the Reason header in a SIP Response may be used by the MMTEL AS to determine subsequent session handling actions.
Record-Route	
Recv-Info	SRVCC for alerting calls will send/receive Recv-Info header containing the applicable g.3gpp.state-and-event package name.
Referred–By	
Refer–To	
Replaces	
Reply–To	
Request-Disposition	Support of the “no-fork” fork-directive.
Require	
Resource-priority	MMTEL AS understands ets and wps namespaces of the Resource-priority header.
Retry–After	
Route	
Rseq	
Server	MMTEL AS can pass through the received value or by configuration insert a Server header in all SIP Responses, with information about the MMTEL AS version.
Session–Expires	
Session-Id	Used for 3PTY service to identify sessions to be used in 3PTY. MMTEL AS adds the Session-Id header if it is not present in the initial INVITE Request and in Responses.
Subject	
Subscription–State	
Supported	
Target-Dialog	SRVCC Release-10 introduces the Target-Dialog header in SCC AS. Target-Dialog header field is used in INVITE(ATUSTI) request coming from ATCF in order to identify the old PS dialog to be transferred.
To	To header may be modified between received and sent initial requests. MMTEL AS generates a new to-tag for each dialog created from an initial request. The To header, including to-tag, is mapped between received and sent responses and subsequent requests in a dialog.
Unsupported	
User–Agent	MMTEL AS can pass through the received value or by configuration insert a User-Agent header in all SIP Requests, with information about the MMTEL AS version.
Via	
Warning	Service specific Warning headers may be included in reject responses.

4.2.4 Bodies

This section specifies the message bodies that are used by MMTEL AS.

Any other message bodies that are received are passed on transparently when the MMTEL AS node is working in B2BUA mode, and are ignored when the MMTEL AS node is working in UAS or UAC mode.

4.2.4.1 Properties

All documents in the body of a SIP method /response can have the following properties

Generic Properties of SIP Documents

Property	Purpose/ Possible values
Content Type3 (MIME subtype)	This will map to either a MIME type or MIME sub-type and will uniquely identify the document type
Direction	Possible Values - Generates only - Receives only - Generates & Receives - Forward
Content Disposition1,3	Possible values - package □ Render □ Session □ Info
Handling (by the recipient)2	Possible Values - Optional - Required (default)
Character Set Encoding	As UTF-8 is backwards compatible with ASCII, it is the usual form of encoding.
Version	Possible values □ . Versioning not supported - All versions - List defined subset

Notes
1: The default Content-disposition should be defined for each specific body type. If no default is defined for a body then 'render' is used as the default. The default Content-disposition is used when no Content-disposition header is included for the body in the SIP message.

2: The default value for the handling parameter is 'required'. The default value is used if there is no Content-disposition header is included for the body or if there is no handling parameter included in the Content-disposition header.

3. Content-type is always included in the headers section of a SIP Request which includes a body part. Content-disposition is optional. If the Content-type is 'multipart/mixed' then separate Content-type and optional Content-disposition headers are included in the body section of the message for each separate body included
.

4.2.4.1.1 Content- Type

Each document type supported over this interface is defined to have a unique MIME type (or a MIME sub-type) which equates to the Content-Type.

Content-Type is a header field that defines the body.

XML Content Types were defined in RFC 3023
Each XML Content-Type should be registered with IANA in accordance with RFC 3688
.

4.2.4.1.2 Version

Table provides properties additional to those shown in previous table, which are applicable to xml bodies only.

Additional properties that apply to XML only

Property	Possible or typical values	
Handling of schema version	Document Header	Possible Values

		- Versioning not supported - Version always present - If absent, assume version [version no.]
	Content-Type (as m-params)	Possible Values - Versioning not supported - Version always present - If absent, assume version [version no.]

Individual document types and their properties are defined in the following subsections.

4.2.4.2 MIME

The MIME mechanism is used to carry documents in the body of SIP methods.
Separate behavior is defined for a SIP body carrying one document and for a SIP body carrying multiple documents

4.2.4.2.1 Single document body

This is the simple case with a single content-type definition.

An example is shown below

```
Content-Type: application/vnd.etsi.aoc+xml; sv="2"
Content-Disposition: render
Content-Length: 246
<CRLF>
<Body goes here>
```

4.2.4.2.2 Multi-part bodies

This case requires

i. An overall content-type definition of “multipart/mixed” to indicate the body contains multiple documents of different types. (RFCs 2045 & 2046)

ii. A content-type definition per document

iii. An optional content-disposition definition per document

iv. A boundary which is a unique string that marks the beginning and end of each content-type definition.

An example is shown below

```
Content-Type: multipart/mixed; boundary="959595"
Content-Length: 896
--959595
Content-Type: application/vnd.3gpp.cw+xml;sv="1"
Content-Disposition: render
<CRLF>
<Body goes here>
--959595
Content-Type: application/vnd.etsi.aoc+xml; sv="2"
Content-Disposition: render
<CRLF>
<Body goes here>
--959595--
```

4.2.4.3 SDP

An extension to the body of a SIP method with SDP has the following attributes:

SDP specific properties

Session Description Protocol		
Content Type (MIME subtype)		application/sdp
Direction		Generates and receives
Content Disposition	Generates	session
	Receives	If not included default disposition of 'session' used
Handling	Generates	Required
	Receives	If not included default handling of'required' used
Character Set Encoding		UTF-8
Version		versions not supported

4.2.4.4 Content

The syntax of the SDP body is as defined in RFC 4566.

4.2.4.5 Advice-of-Charge to the User

AOC Specific Properties

Advice-of-Charge		
Content Type (MIME subtype)	application/vnd.etsi.aoc+xml	
Direction	Generates only	
Content Disposition	Render	
Handling (by the recipient)	not included default value implied (required)	
Character Set Encoding	UTF-8	
Version	1.0 (default) or 2	
Handling of schema version	Document Header	Version always present
	Content-Type (as m-params)	Version always present

4.2.4.5.1 Schema

The XML schema for version 1.0 is defined in 3GPP TS 24.647.
The XML schema for version 2 is defined in TS 183 043.
The version of the body sent from the MMTEL AS will depend upon the sv (schemaversion) parameter for the AOC service in the received Accept header.

4.2.4.6 Communication Waiting

Communication Waiting specific properties

Communication Waiting	
Content Type (MIME subtype)	application/vnd.3gpp.cw+xml
Direction	Generates only

Content Disposition	Included: value – render	
Handling (by the recipient)	Included: value – optional	
Character Set Encoding	UTF-8	
Version	Versioning not supported	
Handling of schema version	Document Header	Versioning not supported
	Content-Type (as m-params)	Versioning not supported

4.2.4.6.1 Schema

The XML schema is defined in 3GPP TS 24.615 [24].

4.2.4.7 Dial Tone Management Notification

Dial Tone Management specific properties

DTM		
Content Type (MIME subtype) (Note 1)	text/xml or application/simservs+xml	
Direction	Generates only	
Content Disposition	not included default value implied (render)	
Handling	not included default value implied (required)	
Character Set Encoding	application/xml Note: This definition maps to UTF-8	
Version	Versioning not supported	
Handling of schema version	Document Header	Versioning not supported
	Content-Type (as m-params)	Versioning not supported

Note 1: The MMTEL AS may be configured to send DTM notification with Content Type of text/xml or application/simservs+xml.

4.2.4.7.1 Schema

MMTEL AS implemented the dial-tone-management global element as defined in the schema as specified Appendix A of TS 183 043.

4.2.4.8 Conference State - NOTIFY

An extension to the body of a SIP NOTIFY method by the Conference service (both ad-hoc and scheduled type) has the following attributes:

Conference specific properties

Conference State - NOTIFY			
Content Type (MIME subtype)		application/conference-info+xml	
Direction	User MMTEL AS	Forwards	
	Conference MMTEL AS	Generates only	
Content Disposition		not included default value implied (render)	
Handling		not included default value implied (required)	
Character Set Encoding		UTF-8	
Version		Versioning not supported	
Handling of schema version		Document Header	Versioning not supported
		Content-Type (as m-params)	Versioning not supported

4.2.4.8.1 Schema

The XML schema is defined in RFC 4575.

There is an exception in case of Scheduled Conference, where the 'entity' attribute of 'conference-info' element contains the XCON identity of the conference room without namespace nomination instead of the SIP URI of the conference session as defined in RFC 4575.

4.2.4.9 Real-time Transfer of Tariff Information

Properties of RTTI document

RTTI		
Content Type (MIME subtype)	application/vnd.etsi.sci+xml	
Direction	Receives only is the usual mode but will forward any document of this type if the SIP session has charging not active	
Content Disposition	not defined default value implied (render)	
Handling	not included default value implied (required)	
Character Set Encoding	application/xml Note: This definition maps to UTF-8	
Version	Not Supported	
Handling of schema version	Document Header	Versioning not supported
	Content-Type (as m-params)	Versioning not supported

4.2.4.9.1 Schema

The XML schema is defined in Annex C of TS 29.658.

4.2.4.10 URI List – Conference/3PTY creation

Properties of URI list document

URI list		
Content Type (MIME subtype)	application/resource-lists+xml	
Direction	Receives only	
Content	Disposition recipient-list	
Handling	not included default value implied (required)	
Character Set Encoding	application/xml Note: This definition maps to UTF-8	
Version	Not Supported	
Handling of schema version	Document Header	Versioning not supported
	Content-Type (as m-params)	Versioning not supported

4.2.4.10.1 Schema

The URI list body uses two schemas:
- resource-lists which is defined in RFC 4826
- copy-control which is defined in RFC 5364

4.2.4.10.2 Service Level Validation

- Multiple URI list bodies are not allowed.

- The maximum number of entries in the URI list is 31.
  (The 3PTY service accepts only 2 entries).

- The URI in each list entry may contain header parameters like Call-Id, From, To and/or Session-id etc.

- The ‘Session-id’ header parameter is required by the 3PTY service.

4.2.4.11 Communication Diversion – Presence Request

Communication Diversion Presence Request specific properties

CDIV – Presence Request		
Content Type (MIME subtype)	application/simple-filter+xml	
Direction	Generates only	
Content Disposition	not defined default value implied (render)	
Handling	not included default value implied (required)	
Character Set Encoding	application/xml Note: This definition maps to UTF-8	
Version	Not Supported	
Handling of schema version	Document Header	Versioning not supported
	Content-Type (as m-params)	Versioning not supported

4.2.4.11.1 Schema

The schema is defined in RFC 4661.

MMTEL AS specifically requests responses with person and activities as defined in RFC 4480.

4.2.4.12 Communication Diversion – Presence Response

Communication Diversion Presence Response specific properties

CDIV Presence Response		
Content Type (MIME subtype)	application/pidf+xml	
Direction	Receives only	
Content Disposition	If not included default value implied (render)	
Handling	If not included default value implied (required)	
Character Set Encoding	application/xml Note: This definition maps to UTF-8	
Version	Not Supported	
Handling of schema version	Document Header	Versioning not supported
	Content-Type (as m-params)	Versioning not supported

4.2.4.12.1 Schema

The schema is defined in RFC 3863.

4.2.4.13 Communication Diversion – MESSAGE

Communication Diversion Message specific properties

CDIV Message	
Content Type (MIME subtype)	text/plain
Direction	Generates only
Content Disposition	not included default value implied (render)
Handling	not included default value implied (required)
Character Set Encoding	UTF-8
Version	Versioning not supported

4.2.4.13.1 Content

The body included in the MESSAGE Request is a plain text message configured by the network operator.

The syntax is as defined in RFCs 2045 & 2046.

4.2.4.14 NOTIFY Body

NOTIFY specific properties

NOTIFY Body		
Content Type		(MIME subtype) message/sipfrag
Direction		Generates and receives
Content Disposition	Generated	not included default value implied (render)
	Received	If not included default disposition of‘render’ used
Handling	Generated	not included default value implied
	Received	If not included default handling of ‘required’ used
Character Set Encoding		UTF-8
Version		Default

4.2.4.14.1 Content

The body contains a SIP fragment giving details of the NOTIFY; e.g. SIP 2.0/100 Trying.
The syntax of the sipfrag body is as defined in RFC 3420.

4.2.4.15 Communication Completion – NOTIFY

Communication Completion specific properties

CC	
Content Type (MIME subtype)	application/call-completion
Direction	Generates and receives
Content Disposition	not included default value implied (render)
Handling	not included default value implied (required)
Character Set Encoding	UTF-8
Version	Versioning not supported

4.2.4.15.1 Content

- The formal syntax is provided in Call Completion for Session Initiation Protocol (SIP); draft-bliss-callcompletion-03..

4.2.4.16 Communication Completion – PUBLISH

Communication Completion specific properties

CC	
Content Type (MIME subtype)	application/pidf+xml
Direction	Generates and receives
Content Disposition	not included default value implied (render)
Handling	not included default value implied (required)
Character Set Encoding	UTF-8
Version	Versioning not supported

4.2.4.16.1 Schema

The schema is defined in RFC 3863.

4.2.4.17 Closed User Group – INVITE

CUG Specific Properties

CUG indication		
Content Type (MIME subtype)	application/vnd.etsi.cug+xml	
Direction	Forwards	
Content Disposition	If not included default disposition of 'render' used	
Handling (by the recipient)	If not included default handling of 'required' used	
Character Set Encoding	UTF-8	
Version	Versioning not supported	
Handling of schema version	Document Header	Versioning not supported
	Content-Type (as m-params)	Versioning not supported

4.2.4.17.1 Content

This body defines the Closed User Group Interlock Code of the subscriber. The schema to be used is defined in 3GPP TS 24.654 10.2.0 Closed User Group (CUG) using IP Multimedia (IM) Core Network (CN) subsystem, Protocol Specification.

4.3 Formal Syntax or Schema

4.3.1 Requests and Responses

The syntax for SIP Requests and Responses as defined in RFC 3261.

Vendor specific extensions are described below.

4.3.1.1 NOTIFY Request

4.3.1.1.1 User Location Indication

The CSCF may send user location info to MMTEL AS in an unsolicited out-of-dialog SIP NOTIFY with the P-Access-Network-Info (PANI) header set accordingly. In order to make it possible for MMTEL AS to correlate the NOTIFY to an active session, in the same NOTIFY the Charging Info Extension to SIP EVENT Package is used.

4.3.1.1.2 Emergency Call Start/Stop Indication

The CSCF may send an unsolicited out-of-dialog NOTIFY message using the emergency call extension to SIP EVENT package to indicate the start or stop of an emergency call.

4.3.2 Headers

The syntax of SIP headers is as defined in RFC 3261.

4.3.2.1 Event Header

4.3.2.1.1 Charging Info Extension

The Charging Info extension to the SIP Event package is used in relation with the unsolicited out-of-dialog SIP NOTIFY sent by the CSCF to pass on access network info to MMTEL AS.
Example:
Event: charging-info; icid=<value>; call-id=<value>; from-tag=<value>; totag=< value>
MMTEL AS will use the value of the icid parameter to find the corresponding call session.

4.3.2.1.2 Emergency Call Extension

The Emergency Call extension to SIP EVENT package is used by the CSCF to inform MMTEL AS through an unsolicited SIP NOTIFY that a subscriber has started or ended an emergency call.

Example 1:
Event: emergencyCall; start
Example 2:
Event: emergencyCall; stop

4.3.2.2 X-AUT Header

The X-AUT header is used by MMTEL AS to transfer Originating Additional User Category (OAUC) in SIP INVITE messages from the originating network to the terminating network and also to transfer Terminating Additional User Category (TAUC) in SIP responses from terminating network to originating network. The AUC specifies the user or mobile service characteristics in more detail.
The X-AUT header holds either the originating or terminating Additional User Category (AUC) and each header contains two categories, for example ‘Cellular telephone service’ and ‘IMT-2000’. The value of the header is interpreted bit-wise and encoded in hexadecimal.
Syntax is based on ABNF grammar:
X-AUT = “X-AUT” HCOLON 1*(2HEXDIG)
Structure of the header content:Possible values are as follows:

8 (byte bit)	7	6	5	4	3	2	1	Octet number
Type of additional user/service information (1)								1st
Additional user/service information (1)								2nd
.								
Type of additional user/service information (n)								2n-1
Additional user/service information (n)								2n

Values for type of additional user/service information:

00000000	Spare
00000001-10000000	Reserved for network specific use
10000001-11111010	Spare
11111011	Type 3 of additional mobile service information
11111100	Type 2 of additional mobile service information
11111101	Type 1 of additional mobile service information
11111110	Type 1 of additional fixed user information
11111111	Spare

Values for type 1 of additional fixed user information:

00000000	Spare
00000001	Train payphone
00000010	Pink (non|NTT payphone)
00000011-11111111	Spare

Values for type 1 of additional mobile service information:

00000000	Spare
00000001	Cellular telephone service
00000010	Maritime telephone service
00000011	Airplane telephone service
00000100	Paging service
00000101	PHS service
00000110-11111111	Spare

Values for type 2 of additional mobile service information:

00000000	Spare
00000001	HiCap method (analog)
00000010	N/J|TACS
00000011	PDC 800 MHz
00000100	PDC 1500 MHz
00000101	N|STAR satellite
00000110	cdmaOne 800 MHz
00000111	Iridium satellite
00001000	IMT|2000
00001001	PHS (fixed network dependent)
00001010-11111111	Spare

Values for type 3 of additional mobile service information:

00000000- 11111111	Reserved for network specific use (eg. for Charging Plan)

5.3 Bodies
The definition of the bodies for each body used by MMTEL AS is as specified in section 4.2.4.

5 Technical specification of SIP NNI VoIP end user interface (IP PBX interface) – with and without SIP registration

Specification of VoIP protocols for PBX connection that can be integrated with IP Centrex is a supplement to the TECHNICAL SPECIFICATION OF SIP UNI VOIP END USER INTERFACE - chapter 3.

5.1 VoIP Platform Business Trunk Specification

Specification follows following standards and documents:

a) TECHNICAL SPECIFICATION OF SIP UNI VOIP END USER INTERFACE - chapter 3.
b) Sibley, C., Gatch, C., "IP PBX/Service Provider Interoperability", sf-draft-twg-IP_PBX_SP_Interop-sibley-sipconnect, March 2006.
c) ETSI TS 182 025: Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Business Trunking, V2.1.1 (2008-09), Available at http://www.etsi.org/tispan/

5.1.1 IP PBX/Service Provider Interoperability (SIP Connect) Compliance

VoIP Platform is compliant with the IP PBX/Service Provider Interoperability Specification with the following exception: Section 12, Enterprise URI Formatting and Addressing Rules Compliance Clarifications.

5.1.1.1 Section 12, Enterprise URI Formatting and Addressing Rules Compliance Clarifications

Section 12, Enterprise URI Formatting and Addressing Rules lists two options for the communication of the PBX's desired PSTN identity to the SIP Application Server.
VoIP Platform does not fully support Option 1 defined in section 12.1.1. When a Privacy header with value "id" is received by VoIP Platform, VoIP Platform restricts the calling line identity for the call. VoIP Platform does not use the From header to obtain the public identity (calling line identity) for the call. However, VoIP Platform identifies the subscriber based on the P-Asserted-Identity header, when present, so the service set invoked is for the subscriber identified by the private identity as per section 12.1.1. Specifically, when Option 1 is used with VoIP Platform, all calls generated by the PBX using this option have their calling line identity restricted. The correct service profile for the subscriber is used but calling line identity restriction is enforced for the call.
VoIP Platform fully supports Option 2 defined in section 12.1.1 using the From header to identify the subscriber.
Section 12.2 describes the format of the To header from the PBX. VoIP Platform does not use the To header to identify the called number so the formatting rules described in section 12.2 are not required.
Section 12.3 describes the format of the To header for emergency calls. VoIP Platform does not use this format. Instead, VoIP Platform uses its internal network database to determine the appropriate public safety answering point (PSAP) for the subscriber.
As per section 12.5, VoIP Platform uses the request-URI to determine the called number. VoIP Platform does not require E.164 format for the called number. VoIP Platform handles both E.164 and non-E.164 numbers in the request-URI and is capable of translating the called number to determine the appropriate destination route.

5.1.1.2 Section 13, Service Provider URI Formatting and Addressing Rules Compliance Clarifications

Section 13, Service Provider URI Formatting and Addressing Rules specifies the SIP Application Server rules for populating the calling line identity and subscriber identity in the INVITE sent to the PBX from the Application Server.

VoIP Platform is fully compliant to Section 13. However, VoIP Platform is completely configurable and can send the calling line identity in the From header in E.164 format (for example, +420405556789), or nationalized format (for example, 405556789).
VoIP Platform is also configurable for its population of the To and Request-URI fields. The format of the user portion of the subscriber identity can be E.164, national number, or even a SIP-Uniform Resource Identifier (URI) (that is, a non-telephone number) if the PBX is able to support SIP-URI PSTN identities.

5.1.2 Parent Registration for the IP PBX or IAD/Gateway Covering All PBX Subscribers

An IP PBX or IAD/gateway shall send a single REGISTER with the contact of the IP PBX or IAD/gateway for each IP PBX number range representing all of the PBX subscribers served by the PBX in appropriate number range. This method is referred to as a parent registration, where the IP PBX or IAD/gateway is the parent registering on behalf of the children, the PBX subscribers.
In this model, VoIP Platform uses the contact from the parent registration for all terminations through the trunk group to the IP PBX or IAD/gateway. The outgoing INVITE from the VoIP Platform trunk group to the IP PBX or IAD/gateway is populated as follows:
- The From header is populated with the calling line identity of the calling party. The From header is populated with anonymous@anonymous.invalid if the calling line identity has been restricted.
- The P-Asserted-Identity header is only included if the device profile associated with the trunk group contains the trusted policy. When the device profile contains the trusted policy, the P-Asserted-Identity header contains the calling line identity of the calling party and the calling party presentation restrictions, if any, are represented in the Privacy header.
- The request-URI is populated with the contact contained in the REGISTER, regardless of the subscriber within the PBX the call is destined for.
- The To header is populated with the identity (address of record) of the PBX subscriber.
- The device profile associated with the business trunk group to support this configuration must have the Registration Capable policy enabled.
- The device profile associated with the business trunk group to support this configuration must also have the Trunk Mode device policy set to Pilot. When this policy is set to Pilot, incoming calls to trunk group users on a registering device use the same contact from the registration for all trunk group users to populate the Request-URI in the SIP INVITE sent to the PBX device. The To header is populated with the trunk user AoR. This means that the user portion of the contact/Request-URI represents the PBX main line/AoR and the To header represents the PBX user line. Note, some PBXs are not able to route on the To header. For example, a business trunking user (member of a trunk group) is configured with the following line/port or identity:

+420333444501@ims.cz

The business trunk device is a registering device and registers with the following contact:
+420333444500@ims.cz
When this policy is enabled, the resulting Request-URI for outgoing INVITEs sent to the PBX for call terminations to the business trunking user is populated with the business trunk device registered contact as follows:
INVITE sip:+420333444500@ims.cz;user=phone;transport=udp SIP/2.0
The To header is populated with the business trunking user (PBX subscriber) as follows:
To:sip:+420333444501@ims.cz;user=phone

5.1.3 Business Trunking PBX Redirection Handling

An IP PBX or IAD/gateway may handle call redirections within the PBX in a variety of ways. For example, a call comes through the business trunk group to a subscriber in the PBX. The PBX subscriber has the Call Forwarding service enabled in the PBX line profile. The call is subsequently forwarded by the PBX back to the PSTN through the business trunk group. The IP PBX or IAD/gateway may handle the forward by sending an INVITE on a new dialog with the request-URI containing the forward destination and a Diversion or History-Info header containing the PBX subscriber identity. Alternatively, the IP PBX or IAD/gateway may send a 3xx response with a contact containing the forward destination and a Diversion or History-Info header containing the PBX subscriber identity. Yet another alternative, the IP PBX or IAD/gateway may send a REFER with the refer-to header containing the forward destination and the referred-by header containing the PBX subscriber identity.
The business trunk interface must support handling redirections on the access-side interface for business trunking and PBX integration to cover all these mechanisms available to an IP PBX or IAD/gateway to handle call redirections. Call redirections from the IP PBX or IAD/gateway on the access-side interface may occur both within and outside an existing dialog. Redirections occurring outside an existing dialog with the business trunk group are typically used by IAD/gateways fronting a TDM PBX. Redirections occurring within an existing dialog with the business trunk group are typically used by IP PBXs using hosted unified communications services that redirect calls to minimize trunking capacity, by removing the IP PBX from the signaling paths for these scenarios. This redirection scenario is common when users on the PBX have their local Call Forwarding service configured to go to a voice portal number that is hosted on VoIP Platform.

5.1.4 In-Dialog Diversion

In-Dialog line-side diversion scenarios are common in business trunking deployments with IP PBXs that conserve SIP trunking resources. For example, consider a scenario where a PSTN originator calls a PBX user, and the PBX user redirects the call before answer to voice mail, to the PSTN, or to a VoIP Platform party. Intelligent PBXs might send a 302 Moved Temporarily response with a Diversion or History-Info header or a REFER request with a Referred-By header to effect the redirection. Alternatively, if the user redirects the call after answer, the PBX might send a REFER request to effect the transfer. In either case, the PBX releases the SIP trunking resource after the redirection succeeds.
If the PBX sends a 302 Moved Temporarily response, the Diversion or History-Info header identifies the redirecting PBX user. If the PBX sends a REFER request, the Referred-By header indicates the redirecting PBX user. In either case, VoIP Platform correctly executes redirecting services for that PBX user. This is shown in Figure 4 “In-Dialog” Diversion. Note that the 302 Moved Temporarily response or REFER request is on the same dialog as the INVITE.

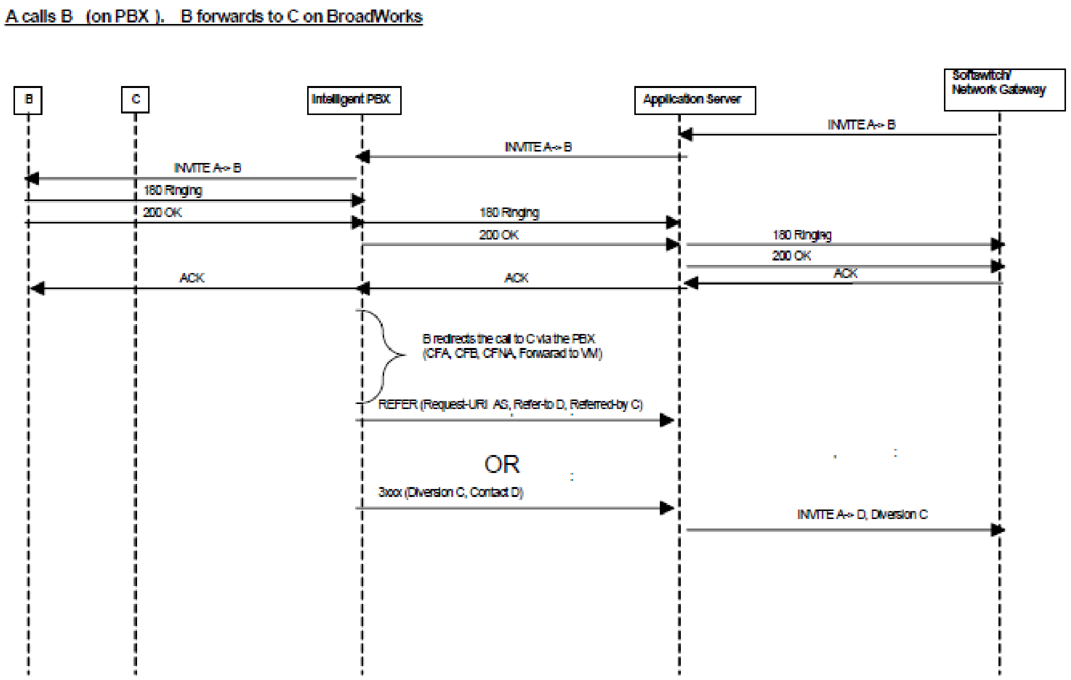

A calls B (on PBX). B forwards to C on BroadWorks
B
C
Intelligent PBX
Application Server
Softswitch/
Network Gateway
INVITE A-> B
INVITE A-> B
INVITE A-> B
180 Ringing
200 OK
180 Ringing
200 OK
180 Ringing
200 OK
ACK
ACK
ACK
B redirects the call to C via the PBX
(CFA, CFB, CFNA, Forwarad to VM)
REFER (Request-URI AS, Refer-to D, Referred-by C)
OR
3xxx (Diversion C, Contact D)
INVITE A-> D, Diversion C

Figure 4 "In-Dialog" Diversion

When a call coming into the intelligent PBX user is subsequently redirected back out of the PBX, the PBX may redirect the call either with a 3xx response or via a REFER to transfer the call to the redirection destination and conserve SIP trunking resources.
Note that if a call comes into the PBX for user B, and user B redirects the call back to VoIP Platform, then the call should be treated as a redirection from the redirecting party identified by the Referred-By header or Diversion/History-Info header depending on the mechanism used to redirect the call. In this scenario, user B's service profile should be processed since it is specified as the redirecting party. However, if user B redirected the call to user C, and user C redirected the call back to the PSTN, to another VoIP Platform subscriber, or to voice mail, then VoIP Platform should use user C's service profile and bypass user B's service profile even though the SIP dialog is associated with user B's session within VoIP Platform. The service profile for user C is processed as if user C redirected the calling party in VoIP Platform. Note that user C is actually redirecting user B; however it is doubtful that the PBXs actually convey this to information to VoIP Platform.

6 Conclusive statements

The end user equipment interoperability shall be tested before massive deployment to guarantee the full functionality of all available features.

7 Annexes (if used)

No annexes

Published as an internal technical standard by O2 Czech Republic a.s.
