GSMA RCS IOT RCS-e Implementation Guidelines

Version 3.2

10 December 2012

.

Security Classification – NON CONFIDENTIAL GSMA MATERIAL

Copyright Notice

Copyright © 2012 **GSM Association**

Antitrust Notice

The information contain herein is in full compliance with the GSM Association's antitrust compliance policy.

Table of Contents

1 Introduction

1.1 Scope

This document provides the highlights of the issues discovered during Interoperability testing (IOT) on the pre-production and production environments of the Operators and contains the guidelines for the Rich Communication Suite-enhanced (RCS-e) related protocols implementation in order to achieve seamless interoperability of RCS-e products and accelerate their time-to-market (TTM).

All clarifications in the current document are related to the latest version of the RCS-e specification [1] available on the GSMA website and all update recommendations of the current document would be incorporated in the new versions of the RCS-e specification.

The guidelines are divided in to six clauses: General and User Interface (UI)/User Experience (UX) issues, Configuration issues, Mobile Operating System (OS) issues, Session Initiation Protocol (SIP)/Session Description Protocol (SDP), Message Session Relay Protocol (MSRP) and Real-Time Protocol (RTP)/Real Time Control Protocol (RTCP) issues. Each clause contains description of issues. These issues are assigned following types:

- Clarification
  Provides further background on functionality already described in the latest version of the RCS-e specification [1] in order to improve understanding.
- Recommendation
  Includes some suggestions on how the functionality required in the latest version of the RCS-e specification [1] can be implemented
- Requirement
  Introduces new requirements that will be included in a future update of the RCS-e specification [1]

The document also includes answers to the frequently asked questions (FAQs).

1.2 Future queries and clarifications

The content of the current document is based on clarification notes provided by the Mobile Network Operators (MNOs) and RCS-e client manufacturers. These notes were collected during the IOT and accreditation processes on the pre-production and production environments and submitted to the GSMA alone or together with the network traces and self-accreditation declaration forms [5], [6]. All the test cases were executed using the RCS-e Test Matrix tool [2]. Detailed information on the IOT and accreditation process could be found in the ‘Guidelines for Licensing Framework’ [3] available on the GSMA website.

The content of the current document is intended to be live and would be updated with new clarifications and recommendations received from the MNOs and RCS-e client manufacturers.

If you are currently passing through the self-accreditation process please collect and document all the discovered issues and provide together with the declaration form or else send them to the GSMA RCS IOT Team (rcsiot@gsm.org). For more details on self-accreditation procedures refer to [4]

1.3 Definition of Terms

Term	Description
ACS	Autoconfiguration Server
APN	Access Point Name
AS	Application Server

B2BUA	Back-to-Back User Agent
CPIM	Common Presence and Instant Messaging
DNS	Domain Name System
EUCR	End User Confirmation Request
FAQs	Frequently asked questions
FQDN	Fully Qualified Domain Name
FT	File Transfer service
FW	Firewall
GPRS	General packet radio service
HSPA	High Speed Packet Access
HTTPS	Hypertext Transfer Protocol Secure
IARI	IMS Application Reference Identifier
IETF	Internet Engineering Task Force
IM	Instant Messaging
IMDN	Instant Message Disposition Notification
IMS	IP Multimedia Subsystem
IOT	Interoperability testing
IP	Internet Protocol
IS	Image Share service
LTE	Long Term Evolution
MCC	Mobile Country Code
MGCF	Media Gateway Controller Function
MNC	Mobile Network Code
MNO	Mobile Network Operator
MSISDN	Mobile Station International Subscriber Directory Number
MSRP	Message Session Relay Protocol
NAT	Network Address Translation
NDA	Non-Disclosure Agreement
NNI	Network-to-Network Interface
OEM	Original Equipment Manufacturer
OMA	Open Mobile Alliance
OS	Operating system
P-CSCF	Proxy Call Session Control Function
PS	Packet Switched domain
Multi-RAB	Multi Radio Access Bearer
RCS	Rich Communications Suite

RCS-e	Rich Communications Suite – enhanced, the launch specification announced at Mobile World Congress 2011 and committed to launch by Deutsche Telekom, Orange, Telecom Italia, Telefonica and Vodafone and further developed in the GSMA
RFC	IETF Requests for Comments
RTCP	Real-Time Transport Control Protocol
RTT	Round-Trip delay Time
RTP	Real-Time Transport Protocol
SBC	Session Border Controller
SDP	Session Description Protocol
SIP	Session Initiation Protocol
TC	Test Case
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TTM	Time-to-market
UA	User Agent
UAC	User Agent Client
UAS	User Agent Server
UDP	User Datagram Protocol
UE	User Equipment
UI	User Interface
UNI	User-to-Network Interface
UX	User eXperience
VS	Video Share service
WAP	Wireless Application Protocol
XML	eXtensible Markup Language

1.4 Document Cross-References

Ref	Document Number	Title
[1]	-	RCS-e specification v1.2.2 (errata)
[2]	RCS IOT 001	RCS-e v1.2 Test Cases Matrix
[3]	RCS IOT 002	Guidelines for licensing framework
[4]	RCS IOT 003	Self-accreditation handbook
[5]	RCS IOT 004	Self-accreditation declaration form provided by network providers
[6]	RCS IOT 005	Self-accreditation declaration form provided by RCS-e client's manufacturers
[7]	-	RCS-e v1.2, User Experience Guidance Document
[8]	-	Rich Communication Suite 5.0 Advanced Communications Services and Clients specification

[9]	IR.74	Video Share Interoperability Specification 1.2
[10]	RFC4575	A Session Initiation Protocol (SIP) Event Package for Conference State, IETF RFC http://tools.ietf.org/html/rfc4575
[11]	RFC3841	Caller Preferences for the Session Initiation Protocol (SIP), IETF RFC http://tools.ietf.org/html/rfc3841
[12]	RFC4122	The Universally Unique IDentifier (UUID) URN Namespace IETF RFC http://tools.ietf.org/html/rfc4122
[13]	TS 24.229	3GPP TS 24.229 Release 10, 3rd Generation Partnership IP multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP) http://www.3gpp.org

2 RCS-e implementation clarifications

2.1 General issues

ID_1_1 Chat user selection mechanism and UX

Type	Recommendation
Related spec [1] clause	3.2.3, Figure 25
Related TC [2] ID	ID_RCSE_7_4_2
Publish date	21.02.2012
Date modified	21.02.2012

Description

When starting a group chat from the chat application, there should be an UX interaction/screen allowing the user to choose the participants. The shown list should show all the RCS-e contacts because without checking with OPTIONS, it is not possible to distinguish whether the users are currently available and performing an OPTIONS query for the whole list will be too time-consuming. When a user is selected, an OPTIONS message must be issued to that individual user. Depending on the response, the UI will show the other party as ready for chat or not.

Please also note that if the UX design is such that the screen is both used to start a 1-2-1 (1 user selected) or a group chat, the confirmations should be shown only if 2 or more users are selected as a 1-2-1 chat can occur anyway even the other party is offline.

ID_1_2 Availability for Video Share on 3G coverage

Void

ID_1_3 Group chat lifecycle

Type	Clarification
Related spec [1] clause	3.2.3
Related TC [2] ID	ID_RCSE_4_4_3
Publish date	01.02.2012
Date modified	13.07.2012

Description

Consistently with the specification, a group chat will be terminated either when:

- The group chat session initiator leaves the chat
- The number of participants is less than 2

In order to provide a consistent experience to the user, when the number of participants in a group chat becomes 2 (e.g. there were more participants but others have already left leaving the initiator of the session plus another participant), the File Transfer button should NOT be again shown.

A Group Chat session may be terminated by inactivity, but it may be restarted at any time by any of the participants according to the procedures specified in ID_4_21_3.

ID_1_4 Call and RCS-e services concurrency

Type	Clarification
Related spec [1] clause	2.7
Related TC [2] ID	n/a
Publish date	21.02.2012
Date modified	21.02.2012

Description

A call is received and a Video Share is taking place from user A to B:

After starting the Video Share the capabilities are exchanged again, so depending on the network coverage and UI capabilities (ability to present a simultaneous Video Share or Video and Image Share) of both A and B, the Image and Video Share will be reported as available or not. If both handsets report Image and Video Share as available then:

- A will NOT be able to initiate another share service until the Video Share session is terminated (i.e. user A should not have the possibility to start a new Image or Video Share)
- B will be able to start an Image or Video Share

Again, after B has started the share service, neither user A or B should be able to initiate further RCS-e share services over a call.

ID_1_5 File Transfer and low storage space scenarios

Type	Clarification
Related spec [1] clause	3.4
Related TC [2] ID	ID_RCSE_5_1_1
Publish date	21.02.2012
Date modified	21.02.2012

Description

When exchanging capabilities and provide the right coverage is in place, the File Transfer service (or Image Share) should be reported as available independently of how much available space is available to store files (even it is full or almost full).

At UI level, the behaviour should comply to the UX guidelines (see [7], that is the receiver should be informed that there is not sufficient storage space to accept the received File Transfer request and if the user accepts the transfer nevertheless, the request should be rejected and the user should be informed that this is not possible. From the protocol level though, if a File Transfer (or Image Share) invitation is received, the receiver’s RCS-e client or implementation should check the available storage space. In case the available space is less than the size of the file, the File Transfer should be automatically rejected (no user interaction).

ID_1_6 Units employed for the File Transfer and Image Share configuration parameters

Type	Clarification
Related spec [1] clause	A.1.4
Related TC [2] ID	ID_RCSE_5_7_1
Publish date	21.02.2012
Date modified	21.02.2012

Description

1. The File Transfer maximum and warning values are defined in the RCS-e specification. If you check the spec (Annex A), the limits are defined in KB
2. The Image Share max value is defined by endorsing the RCS Release 2 managed objects spec (we did not redefine it in RCS-e because it was defined already and we wanted to avoid conflict). Because RCS Release 2 is an older spec, it made sense to define it in Bytes.

ID_1_7 RCS-e terminal implementation/client offline behaviour

Type	Requirement
Related spec [1] clause	
Related TC ID	N/A
Publish date	21.02.2012
Date modified	13.07.2012

Description

When offline, a user can compose and queue messages to be sent as described in section 2.7.2 of the RCS-e spec [1] with the difference that when sending the queued messages when online again, a next message may be sent after receiving any provisional response to the SIP INVITE request, including 100 Trying.

ID_1_8 IM 1-to-1 States

Type	Clarification
Related spec [1] clause	3.2.3 Client Assumptions
Related TC [2] ID	n/a
Publish date	21.02.2012
Date modified	13.07.2012

Description

The following states associated to a 1-2-1 IM messages should be clearly identified at UX level:

- Pending: When the user press ENTER to send the message and provided the user is NOT registered with the IMS core (e.g. offline or airplane mode)
- Sent: a first SIP provisional response is received from the network if the message is sent as part of the INVITE or a MSRP 200 OK is received in case the message was sent over MSRP
- Error: When an error different from 486/487 is received
- Delivered: When receiving the delivery notification
- Read: When receiving the displayed notification

ID_1_9 Image Share optimization via image size reduction

Type	Recommendation
Related spec [1] clause	3.3.7
Related TC [2] ID	n/a
Publish date	21.02.2012
Date modified	21.02.2012

Description

In order to provide the user a seamless experience when transferring images and be aligned with other internet applications providing the service, there is a proposal for a compression mechanism for images which are transmitted using the Image Share service.

ID_1_9_1 Image size reduction algorithm

The recommended approach based on the principle of maximizing the range of devices/resolutions where the image will be displayed with sufficient quality is the following:

- The default scale factor F for the image shall be, F = min(1280/w, 1280/h, 1.0). It shall be noted the w (width) and the h (height) shall be used in pixels for the calculation.
- Please note that if the factor (F) is 1, the next step can be skipped.
- Scale both dimensions by the same factor F (same for width and height so the aspect ratio is maintained).
- Compress as JPG with q=75%
- Compare the new image size with the original, and only offer the possibility to send a resized image if the resulting file is smaller than the original one

ID_1_9_2 Image Share

When a user sends an image to another user the size reduction algorithm will take place. Then if:

- The scale factor (F) of the algorithm is lower than 1, and,
- The result of the compression is a smaller file

The smaller file will be used for the Image Share service. Otherwise, the original file will be used.

Finally, it shall be noted that this process of evaluating whether the size reduction is an option and, if so, the size reduction itself shall happen before the SIP INVITE is sent to the recipient.

ID_1_10 Video bandwidth for Video Share

Void

ID_1_11 Video presentation for Video Share

Type	Clarification
Related spec [1] clause	3.3.3
Related TC [2] ID	ID_RCSE_6_1_3
Publish date	21.02.2012
Date modified	13.07.2012

Description

The aspect ratio of the image shall be preserved when the video is resized to be displayed on the UX according to the screen dimensions.

ID_1_12 RCS-e version 1.2.1 errata regarding sub note 9

Void

2.2 Configuration issues

ID_2_1 FQDN resolution

Type	Clarification
Related spec [1] clause	A.1.5
Related TC [2] ID	ID_RCSE_1_1_1
Publish date	21.02.2012
Date modified	21.08.2012

Description

The FQDN resolution is bearer independent and should be performed by the handset following this process:

1. **Step 1: Autoconfiguration**

   As part of the provisioning process using the autoconfiguration server, the handset gets a FQDN for the P-CSCF.

2. **Step 2: Perform a DNS NAPT SRV query**

   Having obtained the destination domain name the Domain Name System (DNS) is asked to provide matching SIP Server Location Information. One or more NAPTR records may be retrieved and the calling application examines these records to find the best match based on priorities and the desired SIP protocol variant:

```
mnc001.mcc234.3gppnetwork.org. IN NAPTR  50 100 "s" "SIP+D2U"  "" _sip._udp.example.com.
mnc001.mcc234.3gppnetwork.org. IN NAPTR  90 100 "s" "SIP+D2T"  "" _sip._tcp.example.com.
mnc001.mcc234.3gppnetwork.org. IN NAPTR  90 100 "s" "SIPS+D2T" "" _sips._tcp.example.com.
```

   In the above example, “D2U” indicates UDP-based SIP, “D2T” indicates TCP-based SIP, -and “SIPS+D2T” indicates TCP-based encrypted SIP. The presence of these fields indicates what variations of SIP are supported on a given SIP server.

   The "s" flag means the next stage is to look up an "SRV" record.

   Depending on the settings in the XML provided by the autoconfiguration server and the coverage (PS or Wi-Fi), the client will make the choice for the SIP access which they are going to use (SIPoUDP, SIPoTLS or SIPoTCP).

3. **Step 3: Perform a DNS SRV query**

   An example set of SIP server SRV records is as follows:

```
_sip._tcp.example.com.     SRV 0 1 5060    sipserv1.example.com.
_sip._tcp.example.com.     SRV 0 2 5060    sipserv2.example.com.
_sip._udp.example.com.     SRV 0 1 5060    sipserv1.example.com.
_sip._udp.example.com.     SRV 0 2 5060    sipserv2.example.com.
_sips._tcp.example.com.    SRV 0 1 5060    sipserv3.example.com.
_sips._tcp.example.com.    SRV 0 2 5060    sipserv4.example.com.
```

   For each of the variations of the SIP protocols supported the SRV records describe:

   - name of the server;
   - which port number SIP uses; and
   - when there are multiple servers, the weights & priorities to allow rough load balancing.

   The calling network asks the DNS for a SRV record for the host corresponding to the specific service/protocol/domain combination that was returned in Step 2.

If there are multiple records with the same service/protocol/domain combination, the caller must sort the records based on which has the lowest priority. If there is more than one record with the same priority, the RFC 2782 shall apply.

From the SRV record get the corresponding server name.

There is potential flexibility in this step for the destination operator to receive the SIP traffic on different servers depending on the desired variation of the SIP protocol – TCP, UDP, encrypted, unencrypted.

4. **Step 4: DNS A-query**

   For the server name returned in Step 3, do a standard DNS lookup to finds its IP address This is a normal "A" (address) record lookup:

```
sipserv1.example.com.          IN A    101.1.2.3
sipserv2.example.com.          IN A    101.1.2.4
```

This FQDN resolution procedure shall apply each time the network allocates a new IP address to the Device (example: handover 3G to Wi-Fi).

ID_2_2 IMS Account blocking

Void

ID_2_3 Clarification on the roaming APN

Void

ID_2_4 Clarification on HTTP configurations parameters and white listing

Type	Requirement
Related spec [1] clause	2.2.2.1.2
Related TC [2] ID	ID_RCSE_1_1_1
Publish date	13.07.2012
Date modified	13.07.2012

Description

In order to be able to support a white list procedure in the HTTP configuration to only allow the RCS-e/Joyn certified clients, the format of the client_version has been defined and shall be sent accordingly by the clients. Also, the client_vendor and terminal_version length constrains have been relaxed to allow strings up to a maximum of 4 characters and not to mandate a fixed 4 characters length:

client_vendor	String that identifies the vendor providing the RCS client.	Y	String (**4 max**), Case-Sensitive
client_version	String that identifies the RCS client version. **client_version_value = Platform "-" VersionMajor "." VersionMinor** **Platform = Alphanumeric (max 9)** **VersionMajor = Number (2 char max)** **VersionMinor = Number (2 char max)** **Example:** **client_version=RCSAndr-1.0**	Y	String (15 max), Case-Sensitive
terminal_vendor	String that identifies the terminal OEM.	Y	String (**4 max**), Case-Sensitive

Table 1 : Client/terminal identification in HTTPS configuration requests

The white list procedure, if the SP decides to implement it, will be triggered when a client request a HTTP configuration. The HTTP configuration gateway will check if the client_vendor and Platform and VersionMajor parts of the client_version matches one of the certified clients in the white list and also check that the VersionMinor is bigger than the one in the whitelist.

This will allow the OEMs to increase the VersionMinor part without notifying neither the GSMA nor the Operators.

2.3 Mobile OS issues

ID_3_1 Android

ID_3_1_1 Avoiding conflict between two joyn clients on the same device (Android only)

Type	Recommendation
Related spec [1] clause	N/A
Related TC [2] ID	N/A
Publish date	13.07.2012
Date modified	21.08.2012

Description

Note this recommendation applies to Joyn clients (embedded or OTT) and that any joyn value-add service propositions which involve complementing the joyn proposition with additional services or joyn services using alternative platforms are not required to follow the procedures described in this section.

In order to prevent having two joyn clients on the same device and, therefore, negative consequences in the user experience, the following mechanism shall be implemented by both joyn embedded and OTT client implementations.

The mechanism is based on the following principles:

- Identifying Android applications as joyn clients using a Manifest.xml meta-data property
- Identifying if a joyn client is enabled by accessing its Shared Preferences and reading a property from it.
- Accessing a joyn client settings screen by sending an intent using the action defined as a Manifest.xml meta-data property.

ID_3_1_1_1 *Client requirements*

Android joyn clients shall define the following meta-data properties in their Manifest.xml file.

Name	Value	Description
gsma.joyn.client	true	Used to identify the application as an joyn client
gsma.joyn.settings.activity	<String>	Equals to the intent action that be used to start the joyn client settings screen

Table 2: Android joyn client Manifest meta-data properties

Android joyn clients shall define a settings screen activity that can be open by third party applications by using a simple intent which action string is equal to the value of the "*gsma.joyn.settings.activity*" meta-data property. Sending that intent to open the settings screen shall require no permission. Thus, the user decides or not to deactivate the third party application.

The following example illustrates the meta-data that shall be added to the Manifest.xml file, as well as a sample settings screen activity.

```
<application
  android:icon="@drawable/icon"
  android:label="@string/app_name">

        <!-- the following meta-data is used to identify the application as a joyn client -->
        <meta-data
              android:name="gsma.joyn.client"
              android:value="true" />

        <!-- the following meta-data is used to provide the value of the intent action that can be used by other
        applications to start the joyn client settings screen -->
        <meta-data
              android:name="gsma.joyn.settings.activity"
              android:value="com.vendor.product.MyJoynSettingsActivity" />
         <!-- joyn client shall define a settings property such that it can be open by third party applications using
        an intent which action string corresponds to the meta-data value defined above -->
        <activity
              android:name=".MyJoynSettingsActivity">
              <intent-filter>
                     <action
                     android:name="com.vendor.product.MyJoynSettingsActivity" />
                     <category
                     android:name="android.intent.category.DEFAULT" />
              </intent-filter>
        </activity>
```

Table 3 : Android meta-data usage

Joyn clients shall define a publicly readable Shared Preferences using the name "*gsma.joyn.preferences*".

The shared preferences shall be created using the joyn client application context, using the mode MODE_WORLD_READABLE.

The shared preferences shall contain a Boolean property named "*gsma.joyn.enabled*".

This property can have two values:

- True: It will mean that the joyn client is enabled (user switch in settings set to ON) and the application has been provisioned successfully.
- False (default value): It will mean that the joyn client is disabled (user switch in settings set to OFF) or the joyn client has never been provisioned yet.

The joyn client will modify the value of this properties according to the rules defined in the following section.

ID_3_1_1_2 *Client start-up behaviour*

A joyn client which is started for the first time on a device, shall:

- Retrieve the list of installed applications from the PackageManager, and identify existing joyn clients by looking for the Boolean meta-data property named "*gsma.joyn.client*", as defined in the previous section.
- For every joyn clients that are found, the client shall open their shared preferences named "*gsma.joyn.preferences*" and retrieve the Boolean property "*gsma.joyn.enabled*", as defined in the previous section.
- If an existing joyn client is found with the Boolean property "*gsma.joyn.enabled*" set to "*True*", it means that client is already active on the device. The new client shall inform to the user that there is another joyn client already configured in the device and that as a pre-requisite to use this one, it is necessary to disable it. In the same pop-up the possibility to access the joyn settings of the active joyn application (via intent mechanism) shall be offered. The intent action used to open the active joyn client settings screen shall be retrieved by reading its Manifest meta-data property named "*gsma.joyn.settings.activity*".

- If there is no existing joyn client, or that none of them are enabled, the new joyn client may proceed with provisioning and registration. Once the client is successfully provisioned and registered to the network it shall open its own "*gsma.joyn.preferences*" shared preferences and set its own "*gsma.joyn.enabled*" property to "True".
- If the joyn client is disabled (e.g. user switch in settings set to OFF) it shall open its own "*gsma.joyn.preferences*" shared preferences and set its own "*gsma.joyn.enabled*" property to "False".

Please note this start-up behaviour shall also apply when:

- There is an attempt to re-activate the disabled client;
- When the disabled client is re-started.

ID_3_1_2 Avoiding to use the standard port with Android 4.0.3 and 4.0.4

Type	Recommendation
Related spec [1] clause	N/A
Related TC [2] ID	N/A
Publish date	21.08.2012
Date modified	21.08.2012

Description

There have been issues observed with Android versions 4.0.3 and 4.0.4 on some devices. In particular, SIP messages sent via large TCP segments (e.g. >512 bytes) with well-known port 5060 (inbound or outbound without TLS) could not be sent or received. Although with another port (e.g. 5062) or UDP it is possible.

Please see the descriptions of the following android issues ids:

http://code.google.com/p/android/issues/detail?id=34727

http://code.google.com/p/android/issues/detail?id=32736

To avoid this issue it is recommended on the network side to change the DNS records and network setup to use UDP and TCP with another server port, e.g. port 5062.

Note: The protocols ports should be the same for UDP and TCP.

On the RCS-e client side it is recommended to avoid the usage of the standard port 5060 and to set another high port for outbound client connections and in the contact header for inbound connections.

ID_3_2 iOS (Apple)

No specific guidelines so far

ID_3_3 Symbian

No specific guidelines so far

ID_3_4 Windows Phone

No specific guidelines so far

2.4 SIP/SDP issues

ID_4_1 Normalization of MSISDNs

Type	Recommendation
Related spec [1] clause	2.9.3
Related TC [2] ID	ID_RCSE_4_1_14

Publish date	21.02.2012
Date modified	13.07.2012

Description

For outgoing requests no normalization is required for the To header and the Request-URI. The format detailed in section 2.9.3.1 of [1] should be used in case the number is not in international format.

Also, in an outgoing request no normalization is required for the MSISDN in From/P-Preferred-Identity since it will have been provided in the provisioning and during registration in international format already.

For incoming requests the MSISDN in From/P-Asserted-Identity will be in international format unless the international format does not exist for that number and should be matched using the same rules which are used when receiving voice calls.

To avoid issues when roaming though for content sharing it is recommended to use the entry corresponding to that number in the address book in case that is in international format rather than the received Caller-ID.

ID_4_2 Using 486 BUSY HERE instead of 603 DECLINE to avoid simultaneous chat sessions when the receiver is not accepting the chat

Void

ID_4_3 Using SIP MESSAGE to carry display notifications

Void

ID_4_4 Hiding identities in CPIM/IMDN

Void

ID_4_5 Network time for chat

Type	Recommendation
Related spec [1] clause	3.2.2.2
Related TC [2] ID	ID_RCSE_7_1_7
Publish date	21.02.2012
Date modified	13.07.2012

Description

As stated in section 3.2.2.2 of the RCS-e specification [1], the network will insert the correct time into the messages. For sent messages however the only clock available at transmission time is the device's own clock.

It is Messaging Server responsibility to deliver messages in the correct order, so the RCS Client is able to rely on the reception time in order to interleave the incoming and outgoing messages. Please note that the ordering of the messages is phone clock based, the shown message time at the UX shall be the network time (when available) in order to correctly display the time of store and forwarded messages.

ID_4_6 Mandatory character of the request notification for chat

Void

ID_4_7 Handling errors on the receiver's end during chat

ID_4_8 IM race conditions

Void

ID_4_9 CPIM formatting

Type	Recommendation
Related spec [1] clause	3.2.2.2
Related TC [2] ID	ID_RCSE_7_1_7
Publish date	21.02.2012
Date modified	21.02.2012

Description

In order to favour the interoperability, the clients shall follow the RFC 3862 and 5438 but also be flexible enough to handle minor deviations that other clients/handsets may implement. As a reference, we are providing the following recommendations:

ID_4_9_1 RFC 4975

RFC4975 says that content-Type for message/cpim is case insensitive. To maximize interoperability we recommend the message type is set to “message/cpim” all in lowercase characters. Please note this is also applicable in all those other cases in RCS-e where there is a SDP negotiation; the type is always coded in lowercase characters.

ID_4_9_2 RFC 3862

Please note the following example is intentionally missing the IMDN disposition notification. Together with the message, we are including some comments marked in red.

m: Content-type: message/cpim (note that if this is part of a multipart, this will include a Content-Length header after Content-Type. If not, and it is included already at SIP level it is ok)
s: (A blank line in the end can be optional, however we still recommend including it)
h: From: MR SANDERS <im:piglet@100akerwood.com>
h: To: Depressed Donkey <im:eeyore@100akerwood.com>
h: DateTime: 2000-12-13T13:40:00-08:00
h: Subject: the weather will be fine today
h: Subject: lang=fr beau temps prevu pour aujourd'hui
h: NS: MyFeatures <mid:MessageFeatures@id.foo.com>
h: Require: MyFeatures.VitalMessageOption
h: MyFeatures.VitalMessageOption: Confirmation-requested
h: MyFeatures.WackyMessageOption: Use-silly-font (Content-length for full body can be added)
s:(again , this blank line can be optional however we still recommend including it)
e: Content-type: text/xml; charset=utf-8 (charset=utf-8 optional of course, however this encoding is recommended to favour interoperability across different language regions)

Table 4. RFC 3862 recommendations for interoperability

ID_4_9_3 RFC 5438

Please note the following example is focusing on the IMDN disposition and therefore, it covers a as a fragment inside the RFC, Content-type: Message/CPIM is missing, but it should be there as the above example and then a final blank line as recommended). Together with the message, we are including some comments marked in red.

From: Alice <im:alice@example.com>

To: Bob <im:bob@example.com>

NS: imdn <urn:ietf:params:imdn>

imdn.Message-ID: 34jk324j

DateTime: 2006-04-04T12:16:49-05:00

imdn.Disposition-Notification: positive-delivery, negative-delivery (", delivery" here is compulsory)

(blank space needed as per RFC 5438 rectification http://www.rfc-editor.org/errata_search.php?rfc=5438)

Content-type: text/plain (here for example this is part of the body, but the blank line is missing)

Table 5. RFC 5438 recommendations for interoperability (1/2)

As a fragment inside the RFC, Content-type: Message/CPIM is missing, but it should be there as the example from 3862 and then a blank line, if included at SIP level it is ok.

From: Bob <im:bob@example.com>

To: Alice <im:alice@example.com>

NS: imdn <urn:ietf:params:imdn>

imdn.Message-ID: d834jied93rf

(blank space needed as per RFC 5438 rectification http://www.rfc-editor.org/errata_search.php?rfc=5438)

Content-type: message/imdn+xml (here for example this is part of the body, but the blank line is missing compared to RFC3862)

Content-Disposition: notification

Content-length: ...

(This blank line between body headers is compulsory to know where the body content starts)

<?xml version="1.0" encoding="UTF-8"?>

<imdn xmlns="urn:ietf:params:xml:ns:imdn">

<message-id>34jk324j</message-id>

<datetime>2008-04-04T12:16:49-05:00</datetime>

<recipient-uri>im:bob@example.com</recipient-uri>

<original-recipient-uri

Table 6. RFC 5438 recommendations for interoperability (2/2)

ID_4_10 New RCS-e user discovery

Void

ID_4_11 Re-registration required due to an unexpected 403 response

Type	Requirement
Related spec [1] clause	2.2.2.7
Related TC [2] ID	ID_RCSE_2_1_X
Publish date	21.02.2012
Date modified	13.07.2012

Description

Section 2.2.2.7 of the RCS-e specification [1] is only applicable in case no Warning header was included in the 403 Error response.

ID_4_12 E timer duration (RFC 3261)

Type	Recommendation
Related spec [1] clause	B.12
Related TC [2] ID	ID_RCSE_7_1_18

Publish date	21.02.2012
Date modified	04.04.2012

Description

In order to guarantee a decent UX experience and RCS-e stack behaviour particularly when the data bearer is 2G/3G/HSPA, the E timer should be set to a significantly greater value than the T1 timer.

Note that you are using UDP over GPRS, therefore retransmissions are very important but there are 2 different scenarios:

1. <u>INVITE transactions:</u>

   The IMS core network sends instantly a 100 Trying response to stop "A Timer" and avoid useless retransmissions.

2. <u>Non INVITE transactions:</u>

   The IMS core network as a proper B2BUA does not send a 100 Trying. Therefore Options response (takes about 6 seconds that is too much) takes a while (2*RTT+ processing time that is a lot) Therefore retransmission happens based on "E Timer".

Therefore, if the E Timer is set to a value which is similar or smaller than the T1 timer (e.g. A1= 0,5s with T1=0,5s), every time an OPTIONS or MESSAGE request and the response is delayed due to a poor connection quality, there will be at least 3 retransmissions (0,5, 1,5 and 3,5 seconds after the OPTIONS is sent) before the 200OK arrives (6seconds). This should be avoided.

ID_4_13 Concatenation of IARI tags

Void

ID_4_14 Instantaneous offline behaviour when offline due to a re-registration

Type	Recommendation
Related spec [1] clause	2.2.2.1.2
Related TC [2] ID	ID_RCSE_2_1_X
Publish date	21.02.2012
Date modified	21.02.2012

Description

At protocol level, any request that failed with a 403 should trigger a re-register and then re-send the request. However as described in the UX Guidelines document (see [7]), this raises some complications in the rare case where the re-register takes long or fails completely. Since in well-behaved clients this 403 should never happen, there is a proposal to limit handling:

- Failed IM: Queue in persistent storage, send again when re-registered
- Other requests: FT/IS/VS/group chat: Should be retried when the registration is restored with a maximum of 5 seconds. If it takes more than 5 seconds, a message shall be shown to the user suggesting to retry later.

In other words, receiving a 403 will put the client in "offline mode" temporarily until registration is restored.

ID_4_15 1-2-1 to group chat extension

Type	Clarification
Related spec [1] clause	3.2.2.2
Related TC [2] ID	ID_RCSE_7_1_1

Publish date	21.02.2012
Date modified	13.07.2012

Description

The extension of a 1-2-1 chat to a group chat is not used any more. Instead when starting a group chat from a 1-2-1 chat window, a complete independent group chat shall be created.

That is, there is no difference between creating a new group chat and extending a 1-2-1 group chat to a group chat anymore.

ID_4_16 Video interoperability: H264 profile 1b encoding

Void

ID_4_17 SDP in SIP OPTIONS

Void

ID_4_18 Video Share options exchange

Type	Recommendation
Related spec [1] clause	3.3.2
Related TC [2] ID	ID_RCSE_4_1_1
Publish date	21.02.2012
Date modified	16.10.2012

Description

Taking into account the following arguments:

1. Some solutions (particularly clients) are able to detect when the phone is making a call however, it is not possible for them to detect when the call is answered by the other party because the application layer is not providing this information via events
2. As a consequence, the caller implementation does not know when is the right time to send the options causing timing issues:
   a) If sent too early, the receiver does not reply with the VS and IS capabilities because the call is not active
   b) If sent too late, the user experience is bad because the capabilities take a while to appear
3. On the receiver side, it is always possible to capture the event on when the user answers the phone so this problematic is not there.

IR.74 recommends both parties to do the options exchange just because we cannot assume an embedded implementation which is able to access other APIs than any other mobile OS APIs. The following 2 solutions are proposed and acceptable:

- Both (caller and receiver) send OPTIONS
- Only the receiver send OPTIONS because the receiver always knows at all layers when the call is answered and then active

ID_4_19 Group chat participants limit and race conditions

Void

ID_4_20 Optimization on the options exchange during a call

Type	Recommendation

Related spec [1] clause	2.3.1
Related TC [2] ID	ID_RCSE_4_7_2
Publish date	21.02.2012
Date modified	21.02.2012

Description

It was observed that the radio environment after establishing a call is sometimes instable and it takes 1-2 seconds to settle (multi-RAB access) leading to packet lost. Therefore, and to avoid this issue, we recommend to introduce a delay on 2 seconds before the OPTIONS message is issued from the receiver.

Also and to make sure that changing conditions of the radio link (e.g. when the handset handovers to 2G and obviously, there is no possibility to run RCS-e services), a SIP OPTIONS message shall be sent every time the screen becomes active during a call (i.e. when the user takes the phone away from his ear to look the screen [proximity sensor triggered]).

ID_4_21 General clarifications around group chat

Type	Requirement
Related spec [1] clause	3.2.5
Related TC [2] ID	ID_RCSE_7_4_7
Publish date	21.02.2012
Date modified	13.07.2012

Description

The Joyn chat service allows the user to be in a conversation with multiple participants, through a Group Chat functionality that resembles permanent groups. Once a group is created, from a user point of view, it remains available as an operative entity as long as the number of participants keeps above two.

This permanent behaviour is actually built on top of temporary sessions in the network. Since this may require re-establishing the Group session when a member sends a new message it could happen that a participant switches between an 'offline' and an 'online' situation with regards to the Group if he misses any of these re-invitations.

To provide a permanent group experience already before RCS 5.1 functionality including store and forward for Group Chat is available, the clarifications provided in this document have been introduced improving the group chat implementation that was described in RCS-e 1.2 and in the previous clarifications.

Please note that these clarifications provide forward compatibility with the future group chat implementation introduced in RCS 5.1, but do not require an update of the currently available IM servers.

The permanent Group Chat like user experience is achieved by assigning at creation time a globally unique ID to each group chat a unique ID. This ID will be used as Contribution-ID in the group chat sessions. So, when a client receives an incoming group chat session invitation, it will be able to retrieve the corresponding group chat, if any, based on the value of the Contribution-ID provided in the SIP INVITE request.

When user wants to send a message to a group chat, the client shall check if there is an active group chat having the corresponding Contribution-ID value. If there is no active session, the client shall restart the group chat session by establishing a Group Chat session using the participant list that was stored at the end of the last session in which it participated and re-using the same Contribution-ID in the new invitation.

As store and forward as it is defined in RCS 5.1 is not yet available there are two main problems that need to be addressed in this short term solution:

- As the users may be offline when a group chat session is restarted, if a new participant is added to the group chat, the user will not be notified about this new participant. So, the list of participants in each local device may not be up to date.
- As the life cycle of the group chat is not controlled by the network yet, it may happen that two group chat sessions with the same Contribution-ID (i.e. belonging to the same group chat) are created simultaneously.

Until both issues are solved on the network side, the clients are requested to implement safe guards to mitigate the above problems.

The solution requires the client to support having different concurrent group chat sessions in parallel associated to the same group chat.

An example flow of the problem and the proposed solution:

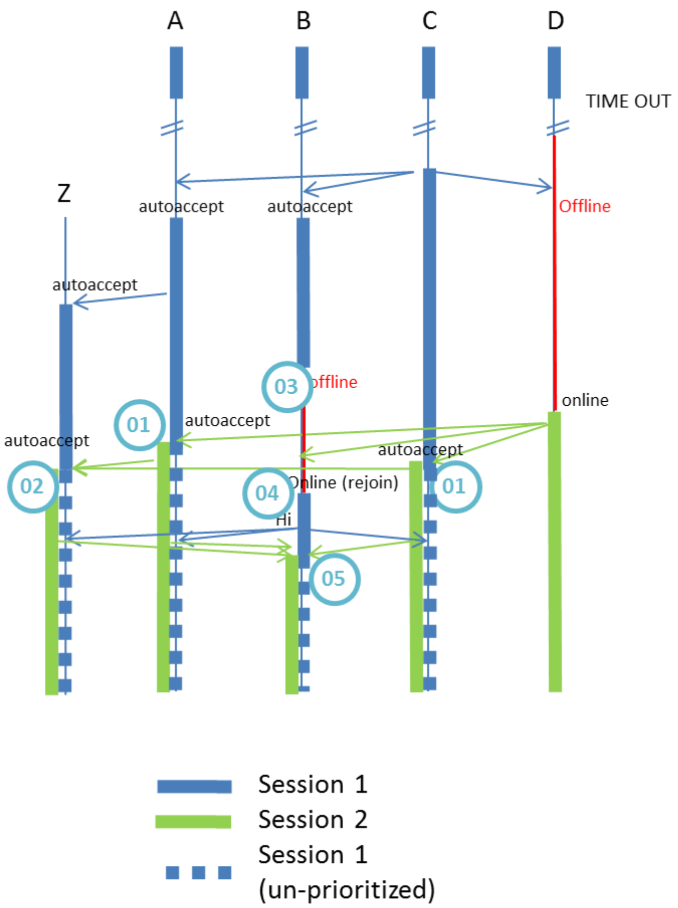

Figure 1 : example group chat concurrency

A group chat has already established between A,B,C and D, and C re-starts the group chat by typing a new message.

D is offline and therefore missed the invitation and the new group chat session. B goes offline due to connectivity losses.

If D becomes online and types a new message to the group chat, it will re-start the group chat by sending a group chat session invitation with the same Contribution-ID.

Once A and C user receives a new group chat session with the same Contribution-ID as an already ongoing one:

1. They will use newest session (session 2) to send all messages in order to let expire the old one (session 1).
2. They will invite to session 2 participants of session 1 not already part of session 2 (Participant Z and B)
3. Because B is offline, he won't accept the invitation, but we make sure all active participants have the new list of participant updated.
4. B is online again. B will rejoin to the active un-prioritized session 1. Write a message.
5. Z, A and C invite B to session 2. So B also un-prioritized session 1.

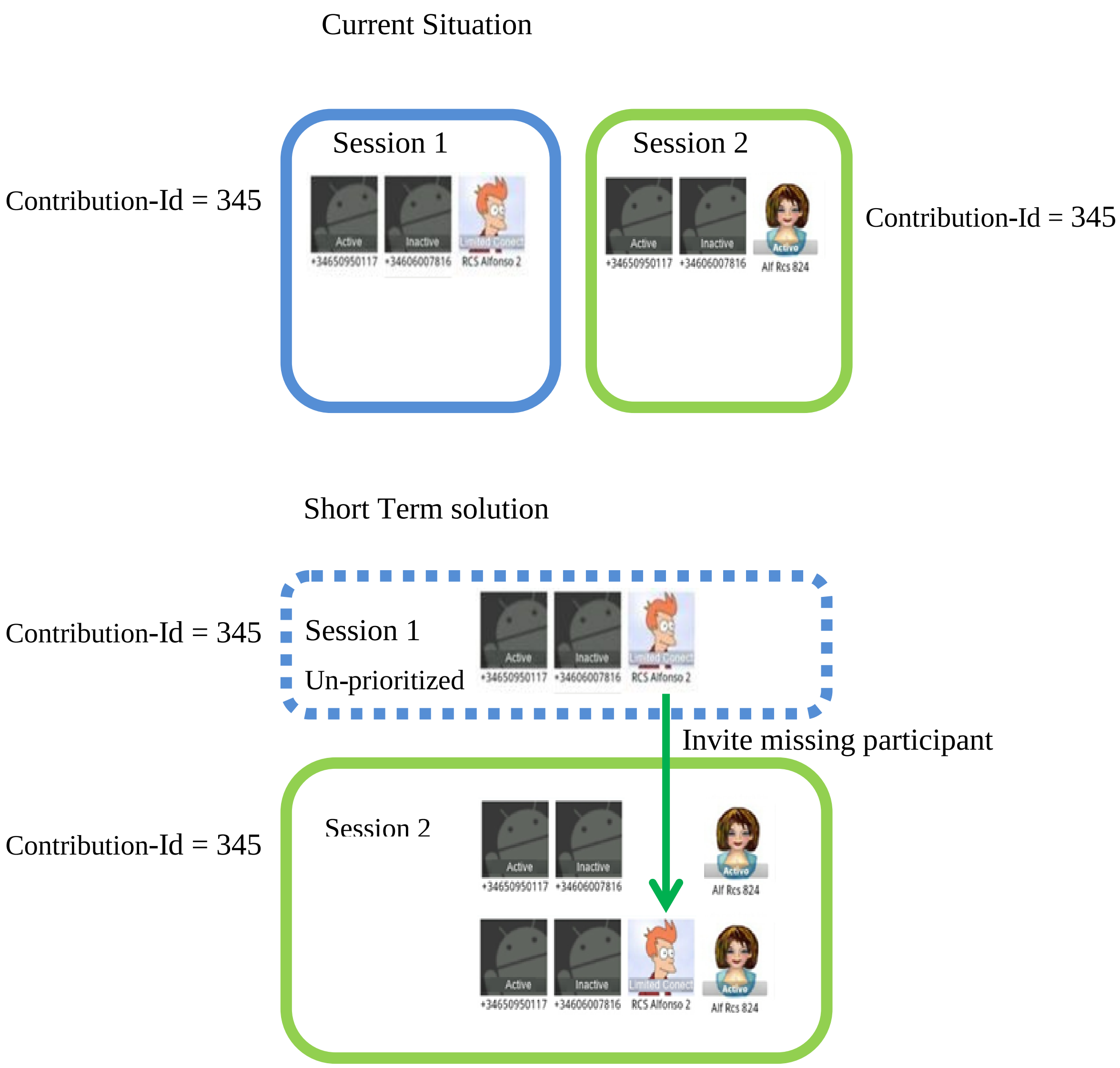

Figure 2: Group Chat Concurrency approach

ID_4_21_1 Clarification on a group chat initiation

In group chat the INVITE sent by the initiating client shall not contain a first message, not in the Subject header nor in a CPIM body.

Note this does not preclude a group chat INVITE including a real subject in the Subject header.

ID_4_21_2 Clarification on a group chat automatic re-join

When the participant (different from the initiator of the session) leaves a group chat involuntarily (e.g. loses data connectivity or a handover between PS and Wi-Fi occurs), the client shall implement a mechanism to retry re-joining the chat once the client is registered again with the IMS core as defined in previous section.

ID_4_21_3 Clarification on a group chat re-start

When a Group Chat has been closed due to inactivity, it may be restarted at any time by any of the participants. In order to do so, the RCS-e client will try to rejoin using the focus Session Identity and same Contribution-ID of the previous Group Chat session. Depending on Service Provider policies, the Group Chat may (e.g. in later RCS releases) be automatically restarted as explained below or a 404 error response will be returned. If a 404 error response is returned the RCS-e client shall initiate a new Group Chat re-using the same Contribution-ID and with latest participant list it has available for the Group Chat to build the URI-list in the SIP INVITE request. If the client is not authorized to (re-) create a group it will receive a 403 Forbidden error from the Messaging server including the warning text set to '127 Service not authorised' as specified in OMA SIMPLE IM. In that case the RCS client shall not create a new group chat with the same Contribution-ID.

It may happen that more than one participant in a Group Chat that was closed because of inactivity will restart the Group Chat at the same time, resulting in two or more conference foci being allocated using the same Contribution-ID. Since rejecting a Group Chat invitation or terminating an ongoing Group Chat session with a SIP BYE request is not possible since it would remove the participant from the Group Chat, the RCS-e client shall:

- If more than one Group Chat invitation is received with the same Contribution-ID, the RCS-e client shall establish or reject all the invitations according to the normal procedures.
- If a Group Chat invitation is received with the same Contribution-ID of an already established Group Chat, the RCS-e device will auto accept the new Group Chat session. The participant list contained in the SIP INVITE request has to be compared with the local participant list and if one or more participants are found in the local list and not present in the incoming SIP INVITE request, the RCS-e client will automatically add those participants to the new Group Chat
- The RCS-e client shall be able to receive all incoming messages from any of the established Group Chat sessions with the same Contribution-ID.
- The RCS-e client will send messages to the Group Chat using only the latest established Group Chat session with the same Contribution-ID. This will allow the rest of Group Chat sessions to time out due to inactivity.
- If the participant explicitly leaves the Group Chat, all the Group Chat sessions with the same Contribution-ID will be terminated by the RCS-e client by sending a SIP BYE request.

ID_4_21_4 Clarification on a abandoning a group chat

A user shall be able to voluntarily abandon a group chat. The technical procedure is based in sending a SIP BYE and, consequently, terminating the MSRP session as per standard session termination procedure.

Please note that if a user voluntarily abandons a group chat, no automatic re-join shall be attempted.

ID_4_21_5 Re-joining or re-starting a chat that the user has previously abandoned voluntarily.

A user who left voluntarily a group chat shall be not able to re-join neither to restart a group chat.

ID_4_21_6 Clarifications on adding participants to a Group Chat

The maximum user participant allowed and the current user count for a running group chat is notified by the focus in the maximum-user-count and user-count elements as defined in RFC4575 (see [10]) when the client subscribes to the conference event package.

Participants may be added providing the maximum-user-count is not reached and the focus's Service Provider policy allows it. If these values are not present in the conference event package or the group chat is not started (i.e. timed out by inactivity) then that the MAX_AD-HOC_GROUP_SIZE configuration parameter may be used instead.

ID_4_21_7 Clarifications on Contribution-ID value

The Contribution-ID is required to be a globally unique value. The value used for the Contribution-ID shall not contain any information that allows identifying the client that generated it (such as an IP Address).

A suggested algorithm for generating the Contribution-ID can be found in http://tools.ietf.org/id/draft-kaplan-dispatch-session-id-03.txt

ID_4_21_8 List of participants

Please note that when restarting a chat the client shall consider the complete list of participants. That includes the complete list of participants which is obtained as part of the INVITE/REFER and any other participant that has been successfully (i.e. he/she has accepted to join the chat) added to the chat.

If a user leaves voluntarily (update on the list of participants), they shall be removed from the list. Note this update is different to the case where a participant timeout. This will be done by extending the information provided according to OMA SIMPLE IM with additional elements and values defined in RFC4575 [10]. More specifically following extensions are provided:

- the "disconnection-method" element can be provided with as values "booted", "departed" and "failed" (see ID_4_21_10 for that last value)
- if the "disconnection-method" is set to "failed" (see ID_4_21_10) also the "disconnection-info" element shall be provided including the "reason" sub-element.

```
…
        </conference-state>
        <users>
              <user entity="tel:+34XXXX" state="partial">
                    <endpoint entity="tel:+34XXXX">
                              <status>disconnected</status>
                              <disconnection-method>departed</disconnection-method>
                    </endpoint>
              </user>
        </users>
</conference-info>
…
```

Table 7. Content of the SIP NOTIFY when the user leaves voluntarily

```
…
        </conference-state>
        <users>
              <user entity="tel:+34XXXX" state="partial">
                    <endpoint entity="tel:+34XXXX">
                              <status>disconnected</status>
                              <disconnection-method>booted</disconnection-method>
                    </endpoint>
              </user>
        </users>
</conference-info>
…
```

Table 8. Content of the SIP NOTIFY when the connection times out

In the timeout case, the participant is still considered part of the participant list.

Protocol	Method	Request-URI
SIP	NOTIFY	Set to the Contact address that the terminating UE has

		registered. The Contact address is normally expressed as a SIP URI.
Header	**Mandatory/ optional**	**The procedure specific values of the parameter**
Subscription-State	M	Indicates status of the subscription (NOTE 1)
Event	M	Conference
Allow-Event	O	Includes a list of tokens which indicates the event packages supported by the server
Content-Type	M	application/conference-info+xml
Content-Length	M	Specifies length of message body
Message body	**Mandatory/ optional**	**The procedure specific values of the parameter**
Body text	M	The message body SHALL contain the Conference state information. XML schema used for NOTIFY messages is described in IETF RFC 4575 [10].

NOTE 1: If the Subscription-State header value is "**active**", it means that the subscription has been accepted and has been authorized. If the header also contains an "expires" parameter, the UE SHOULD take it as the authoritative subscription duration and adjust accordingly. The header value MAY also be "terminated". The "**terminated**" value indicates that the UE SHOULD consider the subscription terminated. In such a case, a reason code MAY also be present. IMS-M never sets the Subscription-State header to a "pending" value.

Table 9. Signalling parameters: SIP NOTIFY request for Conference event

Note: Due to the state-of-art IM-AS design, it is not possible to note participants that have been added to the chat (REFER) who have not yet accepted the chat.

Having said that there is a workaround that RCS clients are expected to implement. If a user adds another user to the chat that is offline, his client shall add it to its local participant list and has the responsibility to re-invite that added user when the chat is restarted again.

ID_4_21_9 Chat autoaccept setting

A new configuration setting is added in order to support group chat autoacceptance and differentiate it from the 1-to-1 chat acceptance. The new parameter name is AutAcceptGroupChat. The details are provided below:

Node: <x>/AutAcceptGroupChat

Leaf node that represent the automatic/manual Group Chat session answer mode

It is not required to be instantiated if a service provider does not enable Group Chat.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	bool	Get

Table 10. IM MO sub tree addition parameters (AutAcceptGroupChat)

- Values: 0, 1
  0- Indicates manual answer mode
  1- Indicates automatic answer mode (default value)
- Post-reconfiguration actions: As the client remains unregistered during configuration, there are no additional actions apart from de-registering using the old configuration and registering back using the new parameter.
- Associated HTTP XML parameter ID: "AutAcceptGroupChat"

And regarding the configuration XML, the only section impacted is the IM one, where the new parameter is added.

```
<characteristic type="IM">
  <parm name="imCapAlwaysON" value="X"/>
  <parm name="imWarnSF" value="X"/>
  <parm name="ftWarnSize" value="X"/>
  <parm name="ftAutAccept" value="X"/>
  <parm name="ChatAuth" value="X"/>
  <parm name="SmsFallBackAuth" value="X"/>
  <parm name="AutAccept" value="X"/>
  <parm name="AutAcceptGroupChat" value="X"/>
  <parm name="MaxSize1to1" value="X"/>
  <parm name="MaxSize1toM" value="X"/>
  <parm name="TimerIdle" value="X"/>
  <parm name="MaxSizeFileTr" value="X"/>
  <parm name="pres-srv-cap" value="X"/>
  <parm name="deferred-msg-func-uri" value="X"/>
  <parm name="max_adhoc_group_size" value="X"/>
  <parm name="conf-fcty-uri" value="X"/>
  <parm name="exploder-uri" value="X"/>
</characteristic>
```

Table 11: Group Chat Auto Acceptance parameter in configuration XML

ID_4_21_10 Clarifications on Closing Group Chat

Any of the participants can close their Chat session associated with an established Group Chat. This can be done from the chat composing window or in the Chat application.

When a participant leaves the Group Chat session with a SIP BYE request, his device unsubscribes from the participant information, and the Group Chat focus will notify the other participants with a new conference state setting that participant‘s state to “disconnected” with disconnection-method “departed”.

Once that Group Chat terminates because of inactivity, that participant who explicitly left cannot rejoin or restart unless he is added by another participant, since that user is no longer on the latest participant list.

When User C closes their Group Chat session the other users will be notified in the chat through a predefined indication “User C has left the conversation”, and their devices will remove him from the displayed recipients. A conversation history will exist in User C’s device history with the messages associated with the chat up to the point the user left.

Any participant can also leave or decline the Group Chat by rejecting the Group Chat invitation with a 603 Decline response. The Group Chat focus will notify the other participants with a new conference state setting the participant state to “disconnected” with disconnection-method “failed” and include in the reason sub-element of the disconnection-info element the code 603 as follows : *[<reason>SIP;cause=603;text="Decline"</reason>].*

This means that in this case, in line with what is described in ID_4_21_8, the disconnection-method and disconnection-info elements defined in [10] shall be provided in the conference state in addition to the information provided according to OMA SIMPLE IM.

An RCS-e client receiving a notification of a participant leaving the Group Chat, either by closing or rejecting the Group Chat session shall remove the participant from the locally stored participant list associated with the Group Chat.

A Group Chat session is closed when

1. less than the minimum active number of participants as defined in the Messaging Server, for a Group Chat remain in the Group Chat, or
2. when a chat inactivity timeout expires, or
3. based on local policy in the Messaging Server, if the originator leaves the Group Chat.

The active participants are the ones in "connected" state or in the "pending" state (i.e. the ones from which a final response has not yet been received).

A participant is removed from the Group Chat participant list either when explicitly leaving the Group Chat session by sending a SIP BYE request or by rejecting the Group Chat invitation with a 603 error response.

The Messaging Server no longer keeps the focus Session Identity for the Group Chat since there is no longer a valid set of participants remaining. Thus, any attempt by a user to join the Group Chat identified by the focus Session Identity will fail.

ID_4_22 File Transfer Termination

Void

ID_4_23 SIP connectivity issues for Clients

Type	Requirement
Related spec [1] clause	N/A
Related TC [2] ID	N/A
Publish date	04.04.2012
Date modified	04.04.2012

Description

It was discovered that there could be problems in MNOs domestic routers if RCS-e clients use the same originating SIP signalling port all the time. To avoid this possible case it is recommended to use a random originating SIP signalling port of the range 1025-65535 in the RCS-e client implementations. If the selected port is not available, the following port number shall be assigned for this session. Mobile OS normally handle this process.

Additionally, to avoid SIP port scanners to drain devices battery or make them malfunction it is recommended that RCS-e clients must reject any SIP traffic not coming from the MNO's SBC or IMS core network.

ID_4_24 Separate session for FT during IM/Chat session

Type	Clarification
Related spec [1] clause	3.4.1
Related TC [2] ID	ID_RCSE_5_5_1
Publish date	04.04.2012
Date modified	04.04.2012

Description

Unlikely to the OMA SIP/SIMPLE IM specifications in the RCS-e specification [1] it is not allowed to start a file transfer with a RE-INVITE during an ongoing IM/Chat session when the corresponding instructions are received.

In order to start a file transfer session during an ongoing IM/Chat session the initiating UAC shall establish separate SIP and MSRP sessions using INVITE request with all required SDP information according to the RCS-e specification [1].

ID_4_25 FT session cancelation by receiver

Void

ID_4_26 Accept-wrapped-types in SDP offer

Void

ID_4_27 Negative IMDN notifications

Void

ID_4_28 Session-Replaces parameter syntax

Void

ID_4_29 Registration procedure intervals

Type	Requirement
Related spec [1] clause	2.1
Related TC [2] ID	ID_RCSE_1_1_1
Publish date	16.05.2012
Date modified	16.05.2012

Description

There should be only one initial REGISTER sent to the network. This initial REGISTER should be sent when the RCS software is ready on the device.

In case of RCS implementation architecture design, if only one REGISTER is not feasible on the device, a minimum interval between two REGISTER must be set to prevent Deny of Service threshold activation. The minimum interval shall be set to 1 second. It should be able to configure this duration via a local parameter on the device.

ID_4_30 INVITEs frequency within S&F procedures

Type	Recommendation
Related spec [1] clause	B.3
Related TC [2] ID	ID_RCSE_7_3_3
Publish date	16.05.2012
Date modified	13.07.2012

Description

In B.3 clause of the RCS specification [1] when User B comes back online the flow shows the server as sending an INVITE for each message stored. The client sends a 180 Ringing for each and when the server sends another INVITE (for the next stored message), the client sends a 180 in response to that INVITE plus a 486 in response to the previous INVITE.

When the store and forward is provided by the terminating network the IM server shall wait for the delivery notification or 180 ringing response of the previously sent message before sending a new message in SIP INVITE request to the same user.

ID_4_31 CPIM body in SIP requests

Void

ID_4_32 File Transfer auto-accept

Type	Clarification
Related spec [1] clause	Table 38
Related TC [2] ID	N/A
Publish date	13.07.2012
Date modified	13.07.2012

Description

In order to increase the success rate for file transfers while the file transfer store and forward functionality is not available, the following functionality shall be added to allow the autoacceptance of files:

- A new RCS-e configuration parameter is added to configure the autoaccept behaviour, ftAutAccept:
    - If set to 1, incoming file transfers shall be autoaccepted provide their size does not exceed the limit imposed by the ftWarnSize configuration parameter.
    - If set to 0, incoming files shall never be autoaccepted.

In addition to this and when ftAutAccept is set to 1, the client shall offer a setting to enable/disable the autoaccept behaviour during roaming. The default value shall be to not autoaccept while roaming.

ID_4_32_1 Additional details on the configuration parameters

Node: <x>/ftAutAccept

Leaf node that describes whether a File Transfer invitation can be automatically accepted

It is not required to be instantiated if a service provider does not enable File Transfer.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	bool	Get, Replace

Table 12: IM MO sub tree addition parameters (ftAutAccept)

- Values:
  0, automatic acceptance is not possible (regardless of the size of the file).
  1, the File Transfer invitation shall be accepted if the size of the file is smaller than the ftWarnSize configuration parameter.
- Post-reconfiguration actions: As the client remains unregistered during configuration, there are no additional actions apart from de-registering using the old configuration and registering back using the new parameter.
- Associated HTTP XML parameter ID: “ftAutAccept”

ID_4_32_2 Warning Size

The parameter *ftWarnSize* indicates the maximum size of the file the device accepts automatically. If the file proposed to be downloaded exceeds the amount of KB define by ftWarnSize, the device SHALL propose a manual acceptation for downloading the file.

Note the detail description is not included because this parameter is already captured in the RCS-e specification version 1.2.2 [1].

ID_4_32_3 Additional details on XML structure

After the change, the only section of the XML which is affected is the IM section, where the new parameters are included:

```
<characteristic type=”IM”>
  <parm name=”imCapAlwaysON” value=”X”/>
  <parm name=”imWarnSF” value=”X”/>
  <parm name=”ftWarnSize” value=”X”/>
  <parm name=” ftAutAccept” value=”X”/>
  <parm name=”ChatAuth” value=”X”/>
  <parm name=”SmsFallBackAuth” value=”X”/>
  <parm name=”AutAccept” value=”X”/>
  <parm name=”AutAcceptGroupChat” value=”X”/>
  <parm name=”MaxSize1to1” value=”X”/>
  <parm name=”MaxSize1toM” value=”X”/>
  <parm name=”TimerIdle” value=”X”/>
  <parm name=”MaxSizeFileTr” value=”X”/>
  <parm name=”pres-srv-cap” value=”X”/>
  <parm name=”deferred-msg-func-uri” value=”X”/>
  <parm name=”max_adhoc_group_size” value=”X”/>
  <parm name=”conf-fcty-uri” value=”X”/>
  <parm name=”exploder-uri” value=”X”/>
</characteristic>
```

Table 13 : File Transfer Auto Acceptance parameter in configuration XML

ID_4_33 Multidevice support

Type	Clarification
Related spec [1] clause	2.15
Related TC [2] ID	ID_RCSE_9_1_x
Publish date	13.07.2012
Date modified	13.07.2012

Description

In order to secure multidevice can be supported in the future and assuming that today there are no networks available to test the gruu functionality, it is assumed that sip.instance mechanism shall be used.

The client shall include a "*sip.instance*" tag, whose value is the instance ID that identifies the user agent instance being registered.

If the RCS client type is embedded and has access to the device IMEI, then sip.instance shall be the IMEI value as per 3GPP TS 24.229 (see [13]). Otherwise, the value of sip.instance shall use either:

- The value provided as part of the device/client configuration in the uuid_Value configuration parameter (see ID_4_33_2). In this case, the network shall follow one of the algorithms described in RFC4122 (see [12]), or,
- If the uuid_Value is not provided as part of the configuration (parameter not present in the configuration or present but with an empty value), the UUID (Universal Unique Identifier) shall be generated as per RFC4122 section 4.2 and in all cases, must not be modified over time.

ID_4_33_1 Additional clarifications on sip.instance usage for multidevice support

When an RCS client is configured to use sip.instance, all SIP requests and responses that contain a Contact header will carry the sip.instance.

When an RCS client is required to ensure that a generated SIP request is sent back to the same device that was identified through sip.instance, a new *Accept-Contact* header is added carrying only the sip.instance tag and instance identifier value as well as the tags explicit and require described in RFC3841 (see [11]).

Regarding the support of routing based on the value of a sip.instance feature tag by an IMS core, there are two possible scenarios:

4. If the IMS core supports the procedures described in RFC3841, then any SIP request with an *Accept-Contact* header that addresses a specific RCS device is only received by that specific instance/device.
5. If not, then all the RCS clients registered using the same IMS identity will receive the SIP request. Consequently, an RCS client supporting the *sip.instance* procedures shall respond to the invite with a 486 BUSY HERE if the identifier value of the *sip.instance* tag included in the *Accept-Contact* header of that incoming SIP request does not match theirs.

ID_4_33_2 Additional details on the configuration parameters

Node: /<x>/Other/uuid_Value

Leaf node that describes a UUID which is required for the sip.instance multidevice approach. In this case the UUID is generated by the Service Provider network following the algorithm described in RFC4122 (see 12]).

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	chr	Get, Replace

Table 14: Other MO sub tree addition parameters (uuid_Value)

- Values: A string containing the UUID value
- Post-reconfiguration actions: As the client remains unregistered during configuration, there are no additional actions apart from de-registering using the old configuration and registering back using the new parameter.
- Associated HTTP XML characteristic type: “uuid_Value”

ID_4_33_3 Additional details on XML structure.

After the change, the section of the XML which is affected is the OTHER section, where the new parameter is included:

```
<characteristic type=”OTHER”>
  <parm name=”endUserConfReqId” value=”X”/>
  <parm name=”allowVSSave” value=”X”/>
  <characteristic type=” transportProto”>
    <parm name=”psSignalling” value=”X”/>
    <parm name=”psMedia” value=”X”/>
    <parm name=”psRTMedia” value=”X”/>
    <parm name=”wifiSignalling” value=”X”/>
    <parm name=”wifiMedia” value=”X”/>
    <parm name=”wifiRTMedia” value=”X”/>
  </characteristic>
  <parm name=”uuid_Value” value=”X”/>
</characteristic>
```

Table 15: UUID parameter in configuration XML

ID_4_341-2-1 chat S&F procedure with different Operators

Type	Clarification
Related spec [1] clause	3.2.4.11, B.4
Related TC [2] ID	ID_RCSE_7_3_x
Publish date	21.08.2012
Date modified	21.08.2012

Description

As the forward action may be initiated from another domain, e.g. as described in section B.4 of the RCS-e specification [1], a client shall only take into account the user portion of the URI received in the P-Asserted-Identity when verifying whether a received SIP INVITE request is for forwarding stored notifications. If the user part of the URI corresponds to ‘rcse-standfw’, the domain part shall therefore be ignored.

ID_4_35 Clarification on including a Reason header

Type	Recommendation
Related spec [1] clause	3.2.2.2
Related TC [2] ID	ID_RCSE_7_4_x
Publish date	10.12.2012
Date modified	10.12.2012

Description

At the minute, the IM-AS implementations Operators have for joyn and joyn hotfixes consider that if hosting a chat (controlling function role) an IM-AS receiving SIP BYE from a user means that the user is voluntarily leaving the chat. The problem encountered during IOT is that the SIP BYE can be generated by the UE (user requested) or by any middle element in the IMS path between the UE and the IM-AS who is hosting the chat (controlling function role). Consequently, Operators got situations where, the SIP BYE is received even the user did not initiate it making the group chat controlling function to remove the user and inform others that he/she has left.

Operators would need to distinguish a SIP BYE coming from a UE during the Group Chat session on the NNI interface, therefore the proposal is that any SIP BYE coming from a UE contains a Reason header that help Operators to distinguish the cases:

- Reason: *SIP;cause=200;text="Call completed"* shall be used when a SIP BYE is initiated from the UE
- If a controlling function receives a SIP BYE request carrying this value for the Reason header field is received in a Group Chat, the Controlling Function shall mark the user as “Departed”. In any other case, the user shall remain as “booted”
- No network element (local/interconnect) shall remove this cause
- Vendors are recommended to add the Reason header in a Group Chat SIP BYE request initiated by the user. It is also recommended for an Operator’s network edge equipment to add the described Reason header if not provided by the clients

Regarding IM-AS, vendors are recommended to have a setting that allows two behaviours both for hotfixes and future releases

- A user is departed ONLY if a SIP BYE with the Reason: *SIP;cause=200;text="Call completed"* headers included, is received.
- A user is departed ONLY if a SIP BYE no reason header is included or Reason: *SIP;cause=200;text="Call completed"* headers included, is received. A SIP BYE with other reason header is received, the user is marked as ‘booted’.

ID_4_36 Clarification on Idle Timer

Type	Recommendation
Related spec [1] clause	Table 67
Related TC [2] ID	ID_RCSE_7_4_x
Publish date	10.12.2012
Date modified	10.12.2012

Description

When hosting a chat on another MNO, his/her own MNO's IM-AS is in the path acting in a role that is called participating function. As a participating function, there are idle timers implemented.

In case of the idle timer for the participating function is smaller than the controlling function one (the IM-AS on the MNO where the chat is being hosted), then the user:

- Will go offline ('booted') and miss messages -> This is if the fix for Reason header described in the ID_4_35 is implemented
- Will appear as leaving the chat -> This is if the fix for Reason header described in the ID_4_35 is NOT implemented

To make sure the timers are not an issue it is recommended to:

- The IM-AS to have separated controlling and participating function idle timers
- The participating function idle timers to be set to a value sufficiently big so the controlling function (IM-AS hosting the chat) rules the idle timer
- The same rule applies to the SPG/P-CSCF MSRP/TCP media timers.
- In the future Operators should agree to a maximum group chat participating function idle timer and SPG/P-CSCF media timers.
- To set the maximum Group Chat idle timer in the controlling function to 300s
- Alternatively, another solution would be to take the IM-AS from the path for group chat in terminating cases, including two triggers:
    - Receiving an invite to participate in a group chat hosted in another network (Controlling function in an external IM-AS)
    - Rejoin to a group chat hosted in another network

2.5 MSRP issues

ID_5_1 MSRP reports

Type	Clarification
Related spec [1] clause	3.2.2.2
Related TC [2] ID	ID_RCSE_7_1_1
Publish date	21.02.2012
Date modified	13.07.2012

Description

The client/handset should either not include the success-report request flag or to include it set to "no".

ID_5_2 MSRPoTLS implementation

Void

ID_5_3 Optimizing File Transfer transfer time

Void

ID_5_4 FT chunk size

Type	Recommendation
Related spec [1] clause	3.4.6
Related TC [2] ID	ID_RCSE_5_1_1
Publish date	21.02.2012
Date modified	21.02.2012

Description

The recommended chunk value size is 10KB.

2.6 RTP/RTCP issues

ID_6_1 RTCP support

Void

ID_6_2 Securing Video Share procedure

Type	Recommendation
Related spec [1] clause	2.8.1
Related TC [2] ID	ID_RCSE_6_1_3
Publish date	21.02.2012
Date modified	16.10.2012

Description

As a reminder and to avoid further confusions and regarding to the symmetric media behaviour described in the RCS-e specification version 1.2, please remember to:

- send periodic dummy RTP at MT side just after receiving the INVITE or after sending 180 ringing, until the receipt of a first incoming RTP packet (recommended rate: 50 to 100ms) and,
- send RTCP keep alive before sending 200 OK.

The sender should allow enough time for the media path to be secured. A default value of 500ms is recommended.

This approach gives more time for the network to perform NAT binding which is beneficial to secure the mentioned binding otherwise there is a big chance the first I-Frame will be lost.

Because the binding has been completed, there is no media (RTP) packets or I-Frame being lost on the way to MT and, thus, MT can start displaying image almost instantly.

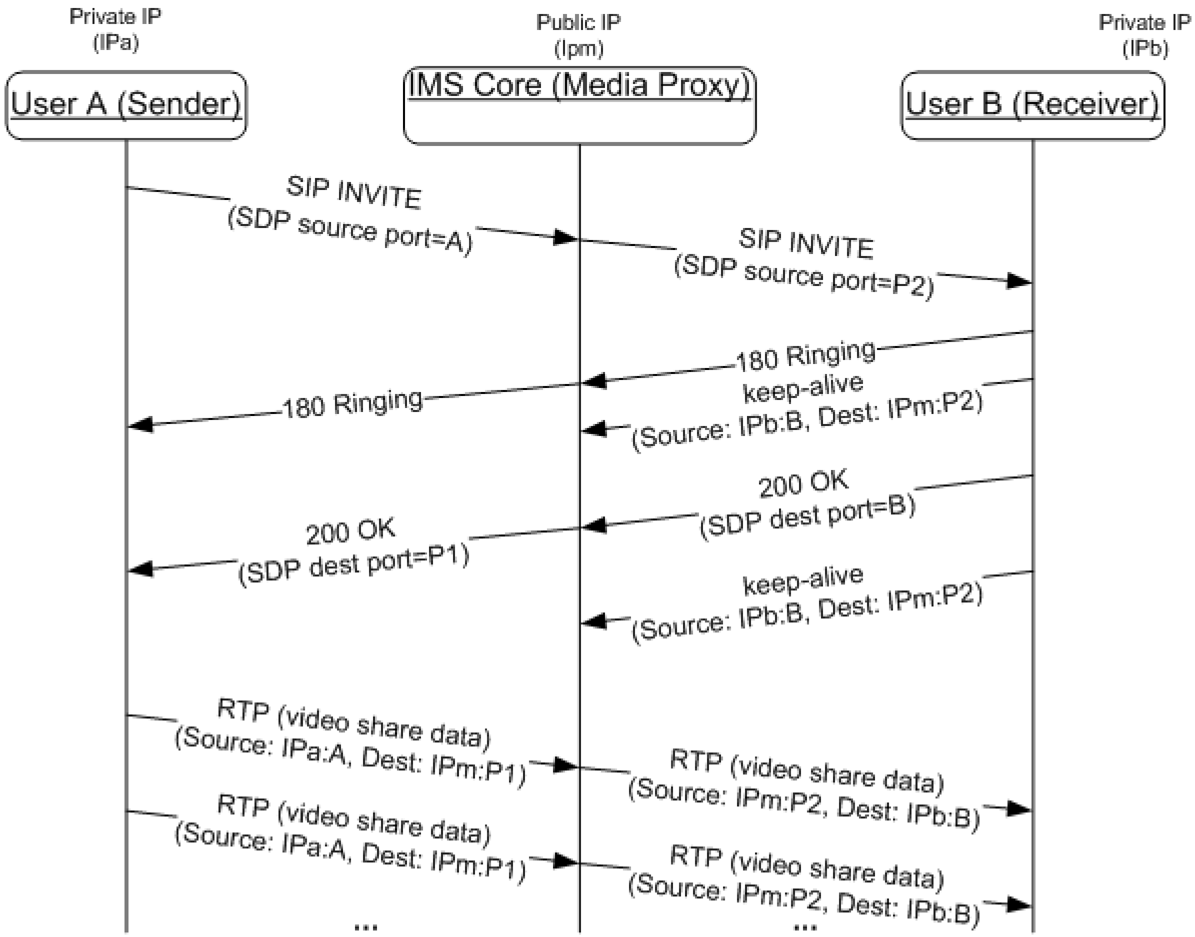

Figure 3: Video Share secured flow

After the receipt of the first incoming RTP packet, it is recommended that the receiver goes on with sending RTP keep-alive, but at a lower rate (15s as default interval is recommended).

Please note that in some circumstances the keep-alive messages may be forwarded on the originating UNI interface. They shall then be transparently ignored by the device.

This approach ensures the compatibility with NAT that need traffic from the internal interface to the external one to maintain the binding.

2.7 End User Confirmation Request (EUCR) issues

ID_7_1 EUCR Clarifications

Void

ANNEX A Frequently asked questions

Q1: What is the expected behavior if TLS/TCP connection gets terminated? Should the client ONLY re-establish the connection OR should the client initiate registration after connection establishment?

The client should re-establish connection. I guess that the same socket will be used, if not reregistration will be needed.

Q2: MSRP: Does the server support sending of the File in ONE chunk?

No problem. IM Server does not limit this. Note that if chunks are big, latency will increase since IM Server does not retransmit the MSRP chunk until it is completely received.

Q3: When should the UE auto-accept a session from the deferred messaging function?

It should accept when P-Asserted-Id is rcse-standfw@domain and only for deferred notifications only (not deferred messages). It will be the a=sendonly session from this PAID with content-type:application/sdp since deferred notifications are sent over MSRP.

Q4: What is the P-Asserted-Identity supposed to be for these 2 scenarios:

Incoming deferred notification:

rcse-standfw@domain.

Incoming deferred IM:

Up to MNO, these messages can be rejected. You will know it is deferred messaging because content-type is multipart/mixed, with a Referred-by header containing the tel-uri (currently is sip-uri but this will be modified today) of the originator, and a PAID that is a different uri.

Q5: Should the UE auto-accept for deferred IM as well?

No, that is why PAID can be different

Q6: Hiding Identities in CPIM / IMDN. This is a new requirement due to security issues over WIFI. Does this apply to messages carrying IMDN only, and not to messages carrying actual text messages?

Both. To avoid dropping of media part over WI-FI (MSRP over TLS is not ready yet) anonymous@anonymous.invalid will work.

Q7: In case of SIM swap, "backup & restore" of Configuration data should be supported. Up to how many SIM cards should be considered?

There is a proposal to support up to 3 SIMs for backup & restore of configuration.

Q8: A clarification for Store and Forward call flow (RCS-e spec, section B.3) is required

- User A is Sending Invite to User B.
- Since User B is offline, Server has accepted the session on behalf of User B.

- User A sends Messages to User B which is stored at server.
- User B comes online, Server start sending Deferred Messages to User B.
- User B Accepts the session and start receiving the stored message from server and send the Delivery and Display notification to server which in turn send the notification to user A.
- After all the stored message has been delivered then server will send the BYE to User B.

Hence, from a Client side handling, we are having difficulty in understanding, what should be the behaviour and when we need to accept-1st call and when we need to accept 2nd incoming call. Are we missing any information that may differ between Session-1 and Session-2 from A's side?

User B at any time may send a new INVITE to user A, and that would cause user A to accept that session and tear down the one it has with the IM Server on behalf of user B. The INVITE will not be rejected with a 486 - it would be the normal procedures where user A's device accepts a new INVITE from the same user, i.e. B, as per b) in section 3.2.4.12 in RCS-e spec:

Device switching (as per the RCS Release 2 OMA-SIMPLE-IM endorsement):

…

If user B changes from one device B1 to another B2 by just sending a new message to the chat from the new device B2. It will send a new INVITE with the message in the subject field as usual that will go to A's device. When A's device detects a new INVITE session from a user (B) which already has an established session it shall end it and accept the new one. All subsequent messages will be received only by device B2. Device B2 must then store the received messages and display them appropriately. If A still has delivery and displayed reports for Device B1, they should be sent before A's device tears down the old session."

Q9: Passing a fingerprint is only for the case using TLS in Peer-to-Peer Mode and there are no service using MSRP in Peer-to-Peer Mode in RCS-e. Should a client support 'fingerprint' mechanism? If yes, should a client support all features including 'Identity' and 'Identity-Info' header fields in RFC 4474?

No, the behaviour of the SBC in MSRP is B2BUA, therefore, the client has only to negotiate with the SBC and the mentioned headers do not need to be supported by the client.

ANNEX B Scope and summary of changes with respect to the previous version

The present version of the RCS-e Implementation Guidelines, 3.0, supersedes the previous version 2.1 which is now considered as deprecated. Consequently all the commercial RCS-e deployments occurring from October 2011 onwards should follow this version of the Guidelines until the new version, superseding the present one is published.

Based on approval of the new RCS-e specification version 1.2.2 that supersedes the version 1.2.1 referred to in the previous version of the guidelines, a number of guidelines were removed that have either incorporated into the new specification and were for that reason no longer required to be included in this document. Next to this a number of clarifications were added or modified clarifications based on the latest developments of the RCS-e services.

As the reference, the main delta between the RCS-e Implementation Guidelines versions 2.1 and 3.0 is listed in the following table:

Clarification # in v2.1	Title	Status
ID_1_2	Availability for Video Share on 3G coverage	Removed
ID_1_3	Group chat lifecycle	Modified
ID_1_7	RCS-e terminal implementation/client offline behaviour	Modified
ID_1_8	IM 1-to-1 States	Modified
ID_1_10	Video bandwidth for Video Share	Removed
ID_1_11	Video presentation for Video Share	Modified
ID_1_12	RCS-e version 1.2.1 errata regarding sub note 9	Removed
ID_2_2	IMS Account blocking	Removed
ID_2_3	Clarification on the roaming APN	Removed
ID_2_4	Clarification on HTTP configurations parameters and white listing	Added
ID_3_1	Avoiding conflict between two joyn clients on the same device	Added
ID_4_2	Using 486 BUSY HERE instead of 603 DECLINE to avoid simultaneous chat sessions when the receiver is not accepting the chat	Removed
ID_4_1	Normalization of MSISDNs	Modified
ID_4_3	Using SIP MESSAGE to carry display notifications	Removed
ID_4_4	Hiding identities in CPIM/IMDN	Removed
ID_4_5	Network time for chat	Modified

ID_4_6	Mandatory character of the request notification for chat	Removed
ID_4_7	Handling errors on the receiver’s end during chat	Removed
ID_4_8	IM race conditions	Removed
ID_4_10	New RCS-e user discovery	Removed
ID_4_11	Re-registration required due to an unexpected 403 response	Modified
ID_4_13	Concatenation of IARI tags	Removed
ID_4_15	1-2-1 to group chat extension	Modified
ID_4_16	Video interoperability: H264 profile 1b encoding	Removed
ID_4_17	SDP in SIP OPTIONS	Removed
ID_4_19	Group chat participants limit and race conditions	Removed
ID_4_21_1	Clarification on a group chat automatic re-join	Modified
ID_4_21_3	Clarification on a group chat re-start	Modified
ID_4_21_5	Re-joining or re-starting a chat that the user has previously abandoned voluntarily	Modified
ID_4_21_6	Clarifications on adding participants to a Group Chat	Added
ID_4_21_7	Clarifications on Contribution-ID value	Added
ID_4_21_8	List of participants	Added
ID_4_21_9	Chat autoaccept setting	Added
ID_4_21_10	Clarifications on Closing Group Chat	Added
ID_4_22	File Transfer Termination	Removed
ID_4_25	FT session cancelation by receiver	Removed
ID_4_26	Accept-wrapped-types in SDP offer	Removed
ID_4_27	Negative IMDN notifications	Removed
ID_4_28	Session-Replaces parameter syntax	Removed
ID_4_30	INVITEs frequency within S&F procedures	Modified
ID_4_31	CPIM body in SIP requests	Removed
ID_4_32	File Transfer auto-accept	Added
ID_4_33	Multidevice support	Added
ID_5_1	MSRP reports	Modified
ID_5_2	MSRPoTLS implementation	Removed

ID_5_3	Optimizing File Transfer transfer time	Removed
ID_6_1	RTCP support	Removed
ID_7_1	EUCR Clarifications	Removed
Annex B	Scope and summary of changes with respect to the previous version	Added

Document Management

Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
1.0	24.02.2012	First Official Version	RCS IOT MNO	Tom Van Pelt / GSMA
2.0	04.04.2012	Editorial changes made and new clarifications added based on issues discovered during IOT on MNO's networks. All changes were approved by RCS IOT MNO Group and presented in the CR# RCSIOTMNO_Doc_10_001rev.1	RCS IOT MNO	Konstantin Savin / GSMA
2.1	16.05.2012	New clarifications added based on issues discovered during IOT on MNO's networks. All changes were approved by RCS IOT MNO Group and presented in the CR# RCSIOTMNO Doc 15_ 003	RCS IOT MNO	Konstantin Savin / GSMA
3.0	13.07.2012	The RCS-e Implementation Guidelines were updated due to approval of the RCS-e specification v1.2.2. The scope and summary of changes with respect to the previous version are presented in the Annex B of the current document	RCS IOT MNO	Tom Van Pelt / GSMA
3.1	21.08.2012	Typo in ID_2_1 with SIPS+D2T description was improved, additional note added to ID_3_1_1_2 on clients start-up behaviour, recommendation ID_3_1_2 provided for use of SIP port in particular Android versions and finally 1-2-1 chat S&F procedure with different Operators clarified in ID_4_34. All changes were approved by RCS IOT MNO Group and presented in the CR# RCSIOTMNO Doc 27_ 001rev1	RCS IOT MNO	Konstantin Savin / GSMA
3.2	10.12.2012	Clarification on Video Share options exchange (ID_4_18) restored, additional clarifications incorporated into the ID_4_21_8 List of participants and ID_4_21_10 Clarifications on Closing Group Chat, additional recommendation provided to ID_6_2 on Video Share procedure, additional recommendations provided in the ID_4_35 for usage of the Reason header and ID_4_36 on	RCS IOT MNO	Konstantin Savin / GSMA

		Idle timer. All changes were approved by RCS IOT MNO Group and presented in the CR## RCSIOTMNO Doc 31_ 001rev1, Doc 32_001rev1, Doc 33_001rev1, Doc 40_001		

Other Information

Type	Description
Document owner	RCS IOT
Editor / Company	Vodafone Group – IOT Group Lead Oscar Gallego