Rich Communication Suite 6.0 Advanced Communications Services and Client Specification

Version 7.0

21 March 2016

This is a Non-binding Permanent Reference Document of the GSMA

Security Classification: Non-confidential

Access to and distribution of this document is restricted to the persons permitted by the security classification. This document is confidential to the Association and is subject to copyright protection. This document is to be used only for the purposes for which it has been supplied and information contained in it must not be disclosed or in any other way made available, in whole or in part, to persons other than those permitted under the security classification without the prior written approval of the Association.

Copyright Notice

Disclaimer

Antitrust Notice

Table of Contents

1 Introduction

1.1 RCS Principles and Vision

RCS (Rich Communication Suite) provides a framework for discoverable and interoperable advanced communication services and detailed specifications for a basic set of advanced communication services.

Figure 1 presents the set of services specified in RCS. All these services can be deployed using a variety of clients on access networks that can be Service Provider controlled or not.

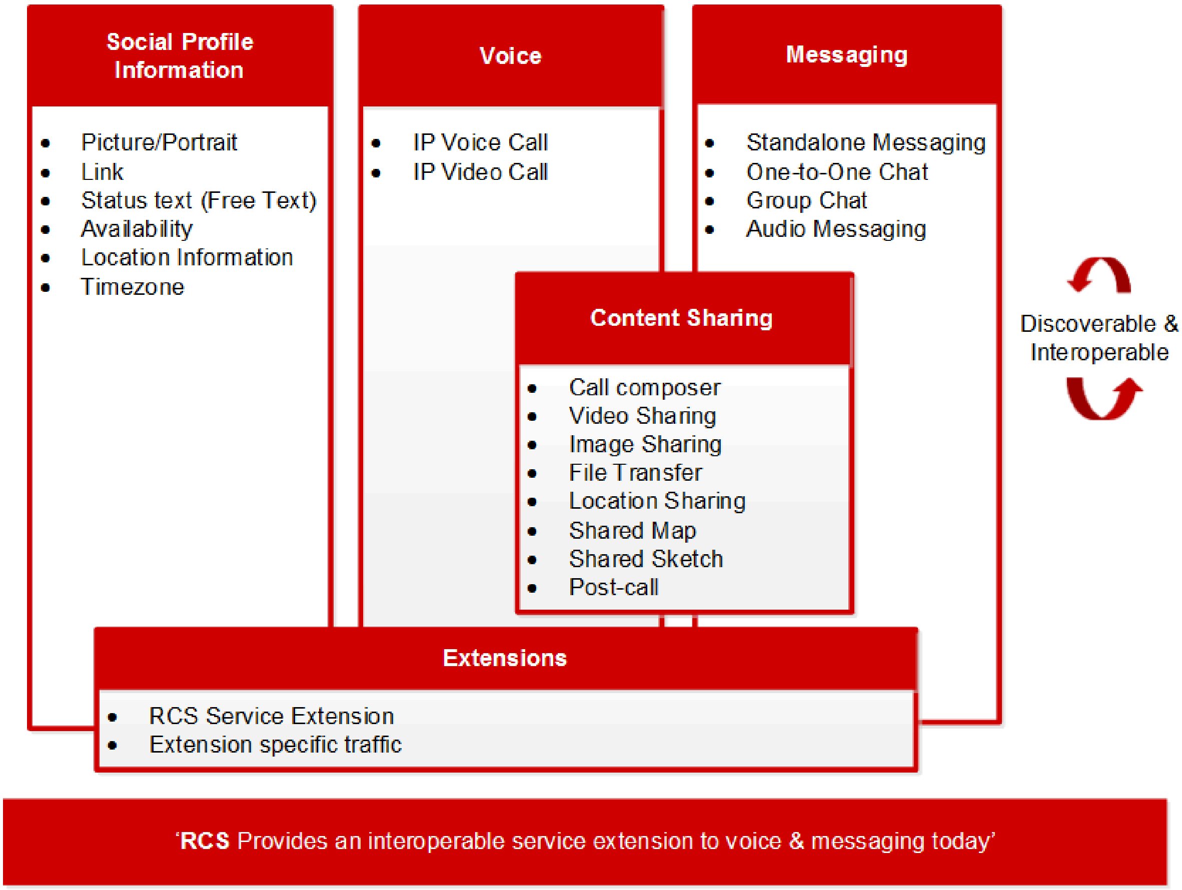

Figure 1: RCS Positioning

As a headline, RCS provides an "interoperable extension to voice and messaging today". The services are designed to run over data networks and can stand alone (e.g. share a picture from the media gallery) or be used in combination with a voice call (e.g. see-what-I-see video).

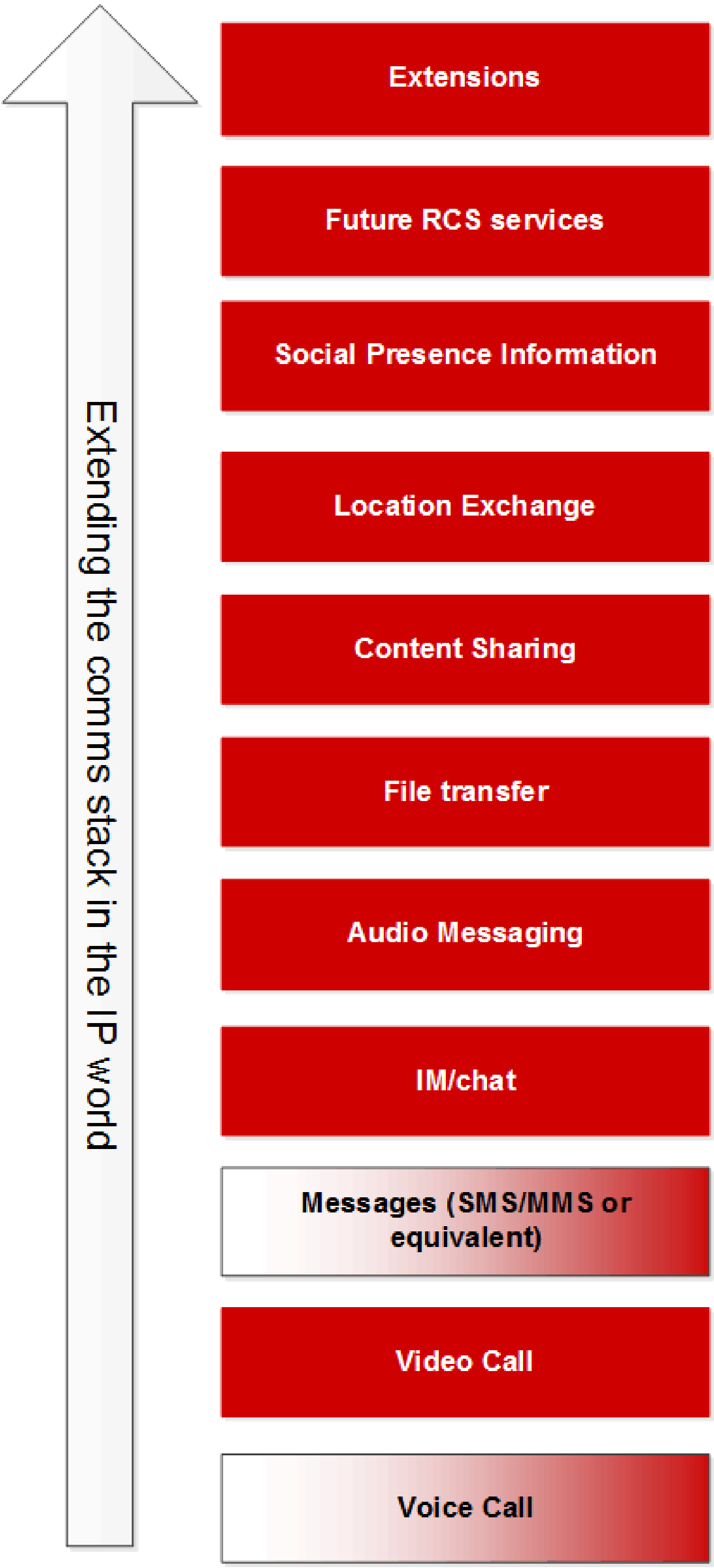

Figure 2: RCS Industry Proposition

As indicated in section 1.2.2, a Service Provider may choose not to deploy all services defined as part of RCS; however when deploying an individual RCS service it will be interoperable with other Service Providers deploying the same service. This also means that even if this specification offers different deployment options to accommodate for different market realities, full interoperability between those deployments is provided for each corresponding service and for the RCS discovery framework.

The cornerstone mechanism that enables RCS is a service or capability discovery framework. For example, when a user scrolls through their address book, they will see their contacts with the RCS services that are available to communicate.

This mechanism is implemented either using the Session Initiation Protocol's (SIP) OPTIONS request or using a Presence-based solution defined in RCS Release 1-4. Both will result in one of three types of response:

1. The contact is registered for service resulting in the contact's current service capabilities being received and logged.
2. The contact is not registered (they are provisioned but not registered).
3. The contact is not found (they are not provisioned for service).

This discovery mechanism is important since it ensures User A can determine what services are available before communicating and allows Service Providers to roll-out new agreed services based on their own deployment schedule or market requirements. These same mechanisms can be used to initially discover (and/or periodically check) the service capabilities of all the contacts within an address book when the user first registers for the service.

1.2 Scope

This document focuses mainly on the User Network Interface (UNI) which to a large extent also determines the Network-Network Interface (NNI). This document also specifies how networks who may choose a different set of deployment options (from the ones described) can work with each other. The interconnect-specific aspects of the NNI are described in a separate document (see [PRD-IR.90]).

It should be noted that the aim of this document is to only specify functionality that can be validated in standard compliant Internet Protocol (IP) Multimedia Subsystem (IMS) pre-production and production environments without major customisation or changes. Service Providers can still introduce customisations and changes to optimise or differentiate their networks however.

It should be noted that all text describing the User Experience (UX), pictures and flow diagrams are for informative purposes only.

1.2.1 Original Equipment Manufacturer (OEM) Integration

This specification is independent from any specific device operating system and is not intended to prescribe the supplier user experience. However, where appropriate key service logic is illustrated through wireframes to aid the reader. It is expected that each device or client supplier will map the basic service principles defined in this document within their own products and drive innovative and differentiated experiences.

1.2.2 Conformance

For terminals, the minimum conformance to this version of the RCS specification can be achieved by a terminal providing the necessary functionality to support the RCS framework, including the capability and new user discovery mechanism (covered in detail in section 2) and one or more of the services specified in detail in section 3. Support for multiple services is optional, however is highly recommended. These conformance criteria ensure that RCS can target low-end devices and therefore boost the market penetration curve.

For networks, the conformance criteria are similar. The framework should be supported including the measures to provide compatibility with all other deployed networks and at a minimum one of the services should be supported. Also, the network should prevent non-compliant clients from connecting to the network or affecting the UNI to a compliant client or the NNI to a compliant network.

1.3 Definition of Terms

Term	Description
2G	2nd Generation of Global System for Mobile Communications (GSM)
ACK	Acknowledgement
ACL	Access Control List
ADSL	Asymmetric Digital Subscriber Line
ALG	Application Layer Gateway
AMR	Adaptive Multi-Rate
APN	Access Point Name
AP	Authentication Proxy
API	Application Programming Interface
AS	Application Server
ASAP	As Soon As Possible
ASO	Arbitrary Slice Ordering
AuC	Authentication Centre
AVC	Advanced Video Codec
BA	Broadband Access
B2BUA	Back-to-Back User Agent
bool	Boolean
BP	Baseline Profile
BPEF	Blacklist Policy Enforcement Function
bps	Bits per second (used with Mbps: Mega-, kbps: kilo-)
BSF	Bootstrapped Security Function
B-TID	Bootstrapping Transaction Identifier
CA	Certification Authority
CAB	Converged Address Book
CBP	Constrained Baseline Profile

Term	Description
CCW	Counter-Clockwise
CPIM	Common Profile for Instant Messaging
CPM	Converged IP Messaging
CRLF	Carriage Return Line Feed
CS	Circuit Switched
CSFB	Circuit Switched FallBack
CVO	Coordination of Video Orientation
CW	Clockwise
DNS	Domain Name System
DNS SRV	Domain Name System Service record
DRX	Discontinuous Reception
DTM	Dual Transfer Mode
DTX	Discontinuous Transmission
e2ae	end-to-access edge
e2e	end-to-end
EAB	Enhanced Address Book
eIMS-AGW	Enhanced IP Multimedia Subsystem-Access GateWay
EOF	End Of File
eP-CSCF	Enhanced Proxy-Call Session Control Function
EPSG	European Petroleum Survey Group
EUCR	End User Confirmation Request
Extension	Piece of software (e.g. add-on, app, etc.) installed on top of an RCS Client that makes use of the RCS infrastructure to change or enhance the user experience or bring extra functionality to the service via the existing or via a new, separate user interface.
FIFO	First IN First Out
FIR	Full Intra Request
FMO	Flexible Macroblock Ordering
FQDN	Fully Qualified Domain Name
FTF	File Transfer Function
FTTH	Fibre To The Home
GAA	Generic Authentication Architecture
GBA	Generic Bootstrapping Architecture
GBR	Guaranteed Bitrate
GGSN	Gateway General Packet Radio Service Support Node
GIF	Graphics Interchange Format
GML	Geography Markup Language
GMLC	Gateway Mobile Location Centre
GPRS	General Packet Radio Service

Term	Description
GPS	Global Positioning System
Group Chat ID	Identifier of a Group Chat. For OMA CPM, it is the Contribution-ID and the Conversation-ID. For OMA SIMPLE IM, it is the Contribution-ID
GRUU	Globally Routable User agent URI
GSM	Global System for Mobile Communications
GSMA	GSM Association
GSO	Group State Object
HOS	Home Operator Services
HPLMN	Home Public Land Mobile Network
HSPA	High Speed Packet Access
HSS	Home Subscriber Server
HTTP	Hyper-Text Transfer Protocol
HTTPS	Hyper-Text Transfer Protocol Secure
HW	HardWare
IARI	IMS Application Reference Identifier
I-CSCF	Interworking Call Session Control Function
ICSI	IMS Communication Service Identifier
ID	IDentifier
IEI	Information Element Identifier
IETF	Internet Engineering Task Force
IM	Instant Messaging. The term chat is also applied in this document to the same concept.
IMAP	Internet Message Access Protocol
IM-AS	Instant Messaging Application Server NOTE: This equivalent terminology for Messaging Server is used in some of the figures
IMDN	Instant Message Disposition Notification
IMEI	International Mobile Station Equipment Identity
IMPI	Internet Protocol Multimedia Subsystem Private Identity
IMPU	Internet Protocol Multimedia Subsystem PUblic identity
IMS	Internet Protocol Multimedia Subsystem
IMSI	International Mobile Subscriber Identity
IMS AKA	IMS Authentication and Key Agreement
Int	Integer
IP	Internet Protocol
IPsec	Internet Protocol Security
IP-SM-GW	Internet Protocol Short Message Gateway
IPX	Internet Protocol Packet eXchange

Term	Description
ISIM	Internet Protocol Multimedia Services SIM
ISF	Interworking Selection Function
IWF	InterWorking Function
JPEG	Joint Photographic Experts Group
KB	KiloByte (i.e. 1024 bytes)
kB	Kilobyte 1 kilobyte = 10^3bytes = 1000bytes.
LBS	Location Based Services
LSB	Least Significant Bit
LTE	Long Term Evolution
MAP	Mobile Application Part
Messaging Server	A server providing support for the standalone messaging service (see section 3.2) according to [RCS-CPM-CONVFUNC-ENDORS] and/or Chat (see sections 3.3 and 3.4) according to [RCS-SIMPLEIM-ENDORS] or [RCS-CPM-CONVFUNC-ENDORS]
MIME	Multipurpose Internet Mail Extensions
MLP	Mobile Location Protocol
MMS	Multimedia Message Service
MMS-C	Multimedia Messaging Service Centre
MMTEL	MultiMedia TELephony
MNO	Mobile Network Operator
MO	Management Object
MO-SMS	Mobile Originated Short Message Service
MPEG	Moving Pictures Experts Group
MSB	Most Significant Bit
MSISDN	Mobile Subscriber Integrated Services Digital Network Number
MSRP	Message Session Relay Protocol
MSRPoTLS	Message Session Relay Protocol over Transport Layer Security
MTU	Maximum Transmission Unit
NAL	Network Abstraction Layer
NAT	Network Address Translation
NGBR	Non-Guaranteed Bitrate
NNI	Network Network Interface
NW	NetWork
OEM	Original Equipment Manufacturer
OMA	Open Mobile Alliance
OMA-CP	Open Mobile Alliance Client Provisioning
OMA-DM	Open Mobile Alliance Device Management
OS	Operating System
PCO	Protocol Configuration Options

Term	Description
P-CSCF	Proxy-Call Session Control Function
PC	Personal Computer
PCC	Personal Contact Card
PDP	Packet Data Protocol
PDF	Portable Document Format
PIDF	Presence Information Data Format
PKI	Public Key Infrastructure
PNB	Personal Network Blacklist
PNG	Portable Network Graphics
POSIX	Portable Operating System Interface
PPS	Picture Parameter Set
PRD	Permanent Reference Document
PS	Packet Switched
PSTN	Public Switched Telephone Network
QCI	Quality of Service Class Identifier
QoS	Quality of Service
RADIUS	Remote Authentication Dial In User Service
RAN	Radio Access Network
RCS	Rich Communication Suite
RCS Group Chat ID	A globally unique identifier that uniquely identifies a Group Chat and that is created when the group chat is first started and preserved over Group Chat restarts. The Group Chat ID is transported as the Contribution-ID header field in a SIMPLE-IM based communication and as the Conversation-ID header field when the communication is CPM based.
RCS User	An end user that has device or client (and the corresponding Service Provider subscription) supporting the RCS capability exchange framework and at least one of the services defined in the current specification.
RFC	Request For Comments
RLC	Radio Link Control
RLS	Resource List Server
RR	Receiver Report
RRAM	RCS Recorded Audio Message
RS	Redundant Slices
RTCP	Real-time Transport Control Protocol
RTP	Real-time Transport Protocol
SBC	Session Border Controller
S-CSCF	Serving Call Session Control Function
SDP	Session Description Protocol
SDES	Session Description Protocol Security Descriptions for Media Streams

Term	Description
SET	Secure User Plane Location Enabled Terminal
SGs interface	3GPP defined reference point between the Mobility Management Entity and the Mobile Switching Centre
SIM	Subscriber Identity Module
SIMPLE	Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions
SIO	Session Info Object
SIP	Session Initiation Protocol
SIPoTLS	Session Initiation Protocol over Transport Layer Security
SLA	Service Level Agreement
SMPP	Short Message Peer-to-Peer
SMS	Short Message Service
SMS-C	Short Message Service Centre
SMSoIP	Short Message Service over Internet Protocol
SP	Service Provider
SPI	Social Presence Information
SPS	Sequence Parameter Set
SR	Sender Report
SRTP	Secure Real-time Transport Protocol
SR-VCC	Single Radio Voice Call Continuity
SSO	Single Sign On
STAP-A	Single-Time Aggregation Packet type A
STUN	Simple Traversal of User Datagram Protocol through Network Address Translations
SUPL	Secure User Plane Location
SW	SoftWare
TCP	Transmission Control Protocol
tel URI	telephone Uniform Resource Identifier
TID	Transaction IDentifier
TLS	Transport Layer Security
TON	Type Of Number
TPDU	Transfer Protocol Data Unit
TUI	(Voice Mail) Telephony User Interface
UA	User Agent
UC	Use Case
UCS2	2-byte Universal Character Set
UDH	User Data Header
UDP	User Datagram Protocol
UE	User Equipment

Term	Description
UI	User Interface
UID	Unique IDentifier
UMTS	Universal Mobile Telecommunications System
UNI	User Network Interface
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
USIM	Universal Subscriber Identity Module
UTC	Coordinated Universal Time
UUID	Universally Unique IDentifier
UX	User Experience
vCard	A format for electronic business cards
VIP	Very Important Person
VMS	Voice Mail System
VoIP	Voice over IP
VoLTE	Voice over Long Term Evolution
VoWiFi	Voice over EPC-integrated Wi-Fi as specified in [PRD-IR.51]
VV-Mail	Visual Voice/Video-Mail
VVM	Visual Voice Mail
W-CDMA	Wideband Code Division Multiple Access
WebRTC	Web Real Time Communication
Wi-Fi	Trademark of Industry Consortium "Wi-Fi Alliance" used as synonym for WLAN (Wireless Local Area Network)
WLAN	Wireless Local Area Network
XCAP	XML Configuration Access Protocol
XDM	XML Document Management
XDMC	XML Document Management Client
XDMS	XML Document Management Server
XML	Extensible Markup Language
XSD	XML Schema Definition
xSIM	Generic reference to different types of SIMs (e.g. USIM, ISIM, etc.)

1.4 Document Cross-References

Ref	Document Number	Title
1	[3GPP TS 23.003]	3GPP TS 23.003 Release 10, 3rd Generation Partnership Project; Numbering, addressing and identification http://www.3gpp.org
2	[3GPP TS 23.038]	3GPP TS 23.038 Release 10, 3rd Generation Partnership Project; Alphabets and language-specific information http://www.3gpp.org

Ref	Document Number	Title
3	[3GPP TS 23.040]	3GPP TS 23.040 Release 10, 3rd Generation Partnership Project; Technical realization of the Short Message Service (SMS) http://www.3gpp.org
4	[3GPP TS 23.140]	3GPP TS 23.140 Release 6, 3rd Generation Partnership Project; Multimedia Messaging Service (MMS);Functional description; Stage 2 http://www.3gpp.org
5	[3GPP TS 23.167]	3GPP TS 23.167 Release 11, 3rd Generation Partnership Project; IP Multimedia Subsystem (IMS) emergency sessions http://www.3gpp.org
6	[3GPP TS 23.221]	3GPP TS 23.221 Release 10, 3rd Generation Partnership Project; Architectural requirements http://www.3gpp.org
7	[3GPP TS 23.228]	3GPP TS 23.228 Release 12, 3rd Generation Partnership Project; IP Multimedia Subsystem (IMS) Stage 2, http://www.3gpp.org
8	[3GPP TS 24.008]	3GPP TS 24.008 Release 12, 3rd Generation Partnership Project, Mobile radio interface Layer 3 specification; Core network protocols http://www.3gpp.org
9	[3GPP TS 24.109]	3GPP TS 24.109 Release 10, 3rd Generation Partnership Project; Bootstrapping interface (Ub) and network application function interface (Ua); Protocol details http://www.3gpp.org
10	[3GPP TS 24.167]	3GPP TS 24.167 Release 10, 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; 3GPP IMS Management Object (MO) http://www.3gpp.org
11	[3GPP TS 24.229]	3GPP TS 24.229 Release 10, 3rd Generation Partnership Project; IP multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP) http://www.3gpp.org
12	[3GPP TS 24.229-rel11]	3GPP TS 24.229 Release 11, 3rd Generation Partnership Project; IP multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP) http://www.3gpp.org
13	[3GPP TS 24.229-rel12]	3GPP TS 24.229 Release 12, 3rd Generation Partnership Project; IP multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP) http://www.3gpp.org
14	[3GPP TS 24.279]	3GPP TS 24.279 Release 8, 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Combining Circuit Switched (CS) and IP Multimedia Subsystem (IMS) services; Stage 3 http://www.3gpp.org

Ref	Document Number	Title
15	[3GPP TS 24.301]	3GPP TS 24.301 Release 11, 3rd Generation Partnership Project; Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3 http://www.3gpp.org
16	[3GPP TS 24.341]	3GPP TS 24.341 Release 10, 3rd Generation Partnership Project; Support of SMS over IP networks; Stage 3 http://www.3gpp.org
17	[3GPP TS 26.114]	3GPP TS 26.114 Release 12, 3rd Generation Partnership Project; IP Multimedia Subsystem (IMS); Multimedia telephony; Media handling and interaction http://www.3gpp.org
18	[3GPP TS 26.141]	3GPP TS 26.141 Release 10, 3rd Generation Partnership Project; IP Multimedia System (IMS) Messaging and Presence; Media formats and codecs http://www.3gpp.org
19	[3GPP TS 33.203]	3GPP TS 33.203 Release 10, 3rd Generation Partnership Project; 3G security; Access security for IP-based services http://www.3gpp.org
20	[3GPP TS 33.220]	3GPP TS 33.220 Release 10, 3rd Generation Partnership Project; Generic Authentication Architecture (GAA);Generic Bootstrapping Architecture (GBA) http://www.3gpp.org
21	[3GPP TS 33.222]	3GPP TS 33.222 Release 10, 3rd Generation Partnership Project; Generic Authentication Architecture (GAA); Access to network application functions using Hypertext Transfer Protocol over Transport Layer Security (HTTPS) http://www.3gpp.org
22	[3GPP TS 33.328]	3GPP TS 33.328 Release 10, 3rd Generation Partnership Project; IP Multimedia Subsystem (IMS) media plane security http://www.3gpp.org
23	[IETF-DRAFT-SIPREC-PROTOCOL]	Session Recording Protocol, Version 02, http://tools.ietf.org/html/draft-ietf-siprec-protocol-02
24	[IETF-DRAFT-RKEEP]	Indication of support for reverse keep-alive, Version 00, June 21, 2012, http://tools.ietf.org/html/draft-holmberg-sipcore-rkeep-00
25	[IETF-DRAFT-RTCWeb_Overview]	Overview: Real Time Protocols for Brower-based Applications, Version 09, 14 February 2014 http://tools.ietf.org/html/draft-ietf-rtcweb-overview-09
26	[PRD-IR.51]	GSMA PRD IR.51 - “IMS Profile for Voice, Video and SMS over Wi-Fi” Version 2.1, 13 August 2015 http://www.gsma.com/
27	[PRD-IR.61]	GSMA PRD IR.61 - “Wi-Fi Roaming Guidelines” Version 10.0, 08 May 2015 http://www.gsma.com/

Ref	Document Number	Title
28	[PRD-IR.64]	GSMA PRD IR.64 - “IMS Service Centralization and Continuity Guidelines” Version 12.0, 15 September 2015 http://www.gsma.com/
29	[PRD-IR.65]	GSMA PRD IR.65 - “IMS Roaming and Interworking Guidelines” Version 17.0, 23 November 2015 http://www.gsma.com/
30	[PRD-IR.67]	GSMA PRD IR.67 - “DNS/ENUM Guidelines for Service Providers & GRX/IPX Providers” Version 11.0, 17 April 2015 http://www.gsma.com/
31	[PRD-IR.74]	GSMA PRD IR.74 - “Video Share Interoperability Specification” Version 2.0, 28 October 2014 http://www.gsma.com/
32	[PRD-IR.79]	GSMA PRD IR.79 - “Image Share Interoperability Specification” Version 2.0, 28 October 2014 http://www.gsma.com/
33	[PRD-IR.88]	GSMA PRD IR.88 - “LTE and EPC Roaming Guidelines” 13.1, 30 June 2015 http://www.gsma.com/
34	[PRD-IR.90]	GSMA PRD IR.90 - “RCS Interworking Guidelines” Version 12.0, 1 April 2015 http://www.gsma.com/
35	[PRD-IR.92]	GSMA PRD IR.92 - “IMS Profile for Voice and SMS” Version 9.0, 8 April 2015 http://www.gsma.com/
36	[PRD-IR.94]	GSMA PRD IR.94 - “IMS Profile for Conversational Video Service” Version 10.0, 22 October 2015 http://www.gsma.com/
37	[PRD-NG.102]	GSMA PRD NG.102 - “IMS Profile for Converged IP Communications” Version 1.0, 19 May 2015 http://www.gsma.com/
38	[RCS-SIMPLEIM-ENDORS]	GSMA PRD RCC.12 RCS 6.0 Endorsement of OMA SIP/SIMPLE IM 2.0, Version 5.0, 21 March 2016 http://www.gsma.com/
39	[RCS-CPM-CONVFUNC-ENDORS]	GSMA PRD RCC.11 RCS 6.0 Endorsement of OMA CPM 2.1 Conversation Functions, Version 5.0, 21 March 2016 http://www.gsma.com/
40	[RCS-CPM-IW-ENDORS]	GSMA PRD RCC.10 RCS 6.0 Endorsement of OMA CPM 2.1 Interworking, Version 5.0, 21 March 2016 http://www.gsma.com/
41	[RCS-3GPP-SMSIW-ENDORS]	GSMA PRD RCC.08 RCS 6.0 Endorsement of 3GPP TS 29.311 Service level Interworking for Messaging Services, Version 5.0, 21 March 2016 http://www.gsma.com/

Ref	Document Number	Title
42	[RCS-CPM-MSGSTOR-ENDORS]	GSMA PRD RCC.09 RCS 6.0 Endorsement of OMA CPM 2.1 Message Storage, Version 6.0, 21 March 2016 http://www.gsma.com/
43	[PRD-RCC.14]	GSMA PRD RCC.14 HTTP-based Service Provider Device Configuration, Version 3.0, 21 March 2016 http://www.gsma.com/
44	[PRD-RCC.15]	GSMA PRD RCC.15 IMS Device Configuration and Supporting Services, Version 2.0, 21 March 2016 http://www.gsma.com/
45	[PRD-RCC.20]	GSMA PRD RCC.20, Enriched Calling Technical Specification, Version 2.0, 21 March 2016 http://www.gsma.com
46	[PRD-RCC.61]	GSMA PRD RCC.61 RCS Common Core 2.0 Service Description Document, Version 3.0, 21 March 2016 http://www.gsma.com
47	[PRIVACY-API]	GSMA Canadian OneAPI Pilot, Privacy Service Developer Guide v1.7, November 2011 https://canada.oneapi.gsmworld.com
48	[PRD-RCC.53]	RCS Device API 1.5.1 Specification, Version 3.0, 23 June 2016 http://www.gsma.com/
49	[VVM]	“Visual Voice Mail Interface Specifications”, Version 1.3, Open Mobile Terminal Platform, OMTP http://www.gsma.com/newsroom/gsmadocuments/
50	[RFC2047]	MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text IETF RFC http://tools.ietf.org/html/rfc2047
51	[RFC2396]	Uniform Resource Identifiers (URI): Generic Syntax IETF RFC http://tools.ietf.org/html/rfc2396
52	[RFC2425]	A MIME Content-Type for Directory Information IETF RFC http://tools.ietf.org/html/rfc2425
53	[RFC2426]	vCard MIME Directory Profile IETF RFC http://tools.ietf.org/html/rfc2426
54	[RFC2595]	Using TLS with IMAP, POP3 and ACAP IETF RFC http://tools.ietf.org/html/rfc2595
55	[RFC2616]	Hypertext Transfer Protocol -- HTTP/1.1 IETF RFC http://tools.ietf.org/html/rfc2616
56	[RFC2617]	HTTP Authentication: Basic and Digest Access Authentication IETF RFC http://tools.ietf.org/html/rfc2617
57	[RFC3261]	SIP (Session Initiation Protocol) IETF RFC http://tools.ietf.org/html/rfc3261

Ref	Document Number	Title
58	[RFC3263]	Session Initiation Protocol (SIP): Locating SIP Servers IETF RFC http://tools.ietf.org/html/rfc3263
59	[RFC3264]	An Offer/Answer Model Session Description Protocol IETF RFC http://tools.ietf.org/html/rfc3264
60	[RFC3326]	The Reason Header Field for the Session Initiation Protocol (SIP) IETF RFC http://tools.ietf.org/html/rfc3326
61	[RFC3329]	Security Mechanism Agreement for the Session Initiation Protocol (SIP) IETF RFC http://tools.ietf.org/html/rfc3329
62	[RFC3428]	Session Initiation Protocol (SIP) Extension for Instant Messaging IETF RFC http://tools.ietf.org/html/rfc3428
63	[RFC3458]	Message Context for Internet Mail IETF RFC http://tools.ietf.org/html/rfc3458
64	[RFC3501]	Internet Message Access Protocol - Version 4rev1, IETF RFC http://tools.ietf.org/html/rfc3501
65	[RFC3550]	RTP: A Transport Protocol for Real-Time Applications, IETF RFC http://tools.ietf.org/html/rfc3550
66	[RFC3711]	The Secure Real-time Transport Protocol (SRTP), IETF RFC http://tools.ietf.org/html/rfc3711
67	[RFC3840]	Indicating User Agent Capabilities in the Session Initiation Protocol (SIP), IETF RFC http://tools.ietf.org/html/rfc3840
68	[RFC3841]	Caller Preferences for the Session Initiation Protocol (SIP), IETF RFC http://tools.ietf.org/html/rfc3841
69	[RFC3858]	An Extensible Markup Language (XML) Based Format for Watcher Information, IETF RFC http://tools.ietf.org/html/rfc3858
70	[RFC3862]	Common Presence and Instant Messaging (CPIM): Message Format, IETF RFC http://tools.ietf.org/html/rfc3862
71	[RFC3863]	Presence Information Data Format (PIDF), IETF RFC http://tools.ietf.org/html/rfc3863
72	[RFC3903]	Session Initiation Protocol (SIP) Extension for Event State Publication IETF RFC http://tools.ietf.org/html/rfc3903
73	[RFC3966]	The tel URI for Telephone Numbers, IETF RFC http://tools.ietf.org/html/rfc3966
74	[RFC3986]	Uniform Resource Identifier (URI): Generic Syntax, IETF RFC http://tools.ietf.org/html/rfc3986

Ref	Document Number	Title
75	[RFC4028]	The Session Timers in the Session Initiation Protocol (SIP), IETF RFC http://tools.ietf.org/html/rfc4028
76	[RFC4122]	The Universally Unique IDentifier (UUID) URN Namespace, IETF RFC http://tools.ietf.org/html/rfc4122
77	[RFC4479]	A Data Model for Presence, IETF RFC http://tools.ietf.org/html/rfc4479
78	[RFC4480]	RPID: Rich Presence Extensions to the Presence Information Data Format (PIDF), IETF RFC http://tools.ietf.org/html/rfc4480
79	[RFC4483]	A Mechanism for Content Indirection in Session Initiation Protocol (SIP) Messages, IETF RFC http://tools.ietf.org/html/rfc4483
80	[RFC4568]	Session Description Protocol (SDP) Security Descriptions for Media Streams, IETF RFC http://tools.ietf.org/html/rfc4568
81	[RFC4572]	Connection-Oriented Media Transport over the Transport Layer Security (TLS) Protocol in the Session Description Protocol (SDP), IETF RFC http://tools.ietf.org/html/rfc4572
82	[RFC4575]	A Session Initiation Protocol (SIP) Event Package for Conference State, IETF RFC http://tools.ietf.org/html/rfc4575
83	[RFC4589]	Location Types Registry, IETF RFC http://tools.ietf.org/html/rfc4589
84	[RFC4648]	The Base16, Base32, and Base64 Data Encodings, IETF RFC http://tools.ietf.org/html/rfc4648
85	[RFC4825]	The Extensible Markup Language (XML) Configuration Access Protocol (XCAP), IETF RFC http://tools.ietf.org/html/rfc4825
86	[RFC4826]	Extensible Markup Language (XML) Formats for Representing Resource Lists, IETF RFC http://tools.ietf.org/html/rfc4826
87	[RFC4867]	RTP Payload Format and File Storage Format for the Adaptive Multi-Rate (AMR) and Adaptive Multi-Rate Wideband (AMR-WB) Audio Codecs, IETF RFC http://tools.ietf.org/html/rfc4867
88	[RFC4961]	Symmetric RTP / RTP Control Protocol (RTCP), IETF RFC http://tools.ietf.org/html/rfc4961
89	[RFC4975]	The Message Session Relay Protocol (MSRP), IETF RFC http://tools.ietf.org/html/rfc4975
90	[RFC5104]	Codec Control Messages in the RTP Audio-Visual Profile with Feedback (AVPF), IETF RFC http://tools.ietf.org/html/rfc5104

Ref	Document Number	Title
91	[RFC5196]	Session Initiation Protocol (SIP) User Agent Capability Extension to Presence Information Data Format (PIDF), IETF RFC http://tools.ietf.org/html/rfc5196
92	[RFC5285]	A General Mechanism for RTP Header Extensions, IETF RFC http://tools.ietf.org/html/rfc5285
93	[RFC5322]	Internet Message Format IETF RFC http://tools.ietf.org/html/rfc5322
94	[RFC5365]	Multiple-Recipient MESSAGE Requests in the Session Initiation Protocol (SIP), IETF RFC http://tools.ietf.org/html/rfc5365
95	[RFC5438]	Instant Message Disposition Notification (IMDN), IETF RFC http://tools.ietf.org/html/rfc5438
96	[RFC5438Errata]	Instant Message Disposition Notification (IMDN), IETF RFC 5438 Errata ID 3013 http://www.rfc-editor.org/errata_search.php?rfc=5438 (see also section C.2)
97	[RFC5491]	GEOPRIV Presence Information Data Format Location Object (PIDF-LO) Usage Clarification, Considerations, and Recommendations, IETF RFC http://tools.ietf.org/html/rfc5491
98	[RFC5547]	A Session Description Protocol (SDP) Offer/Answer Mechanism to Enable File Transfer, IETF RFC http://tools.ietf.org/html/rfc5547
99	[RFC5626]	Managing Client-Initiated Connections in the Session Initiation Protocol (SIP), IETF RFC http://tools.ietf.org/html/rfc5626
100	[RFC5627]	Using Globally Routable User Agent URIs (GRUUs) in the Session Initiation Protocol (SIP), IETF RFC http://tools.ietf.org/html/rfc5627
101	[RFC5819]	IMAP4 Extension for Returning STATUS Information in Extended LIST, IETF RFC http://tools.ietf.org/html/rfc5819
102	[RFC6135]	Alternative Connection Model for the Message Session Relay Protocol (MSRP), IETF RFC http://tools.ietf.org/html/rfc6135
103	[RFC6223]	Indication of Support for Keep-Alive, IETF RFC http://tools.ietf.org/html/rfc6223
104	[RFC6265]	HTTP State Management Mechanism, IETF RFC http://tools.ietf.org/html/rfc6265
105	[RFC6455]	The WebSocket Protocol, IETF RFC http://tools.ietf.org/html/rfc6455
106	[RFC6665]	SIP-Specific Event Notification, IETF RFC http://tools.ietf.org/html/rfc6665

Ref	Document Number	Title
107	[RFC7118]	The WebSocket Protocol as a Transport for the Session Initiation Protocol (SIP), IETF RFC http://tools.ietf.org/html/rfc7118
108	[RFC7162]	IMAP Extensions: Quick Flag Changes Resynchronization (CONDSTORE) and Quick Mailbox Resynchronization (QRESYNC), IETF RFC http://tools.ietf.org/html/rfc7162
109	[RFC7230]	Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing, IETF RFC http://tools.ietf.org/html/rfc7230
110	[RFC7701]	Multi-party Chat Using the Message Session Relay Protocol (MSRP), IETF RFC http://tools.ietf.org/html/rfc7701
111	[GML3.1.1]	OpenGIS® Geography Markup Language (GML) Implementation Specification, Version 3.1.1, OGC 03-105r1 http://www.opengeospatial.org/
112	[CAB_TS]	OMA Converged Address Book (CAB) Specification, Approved Version 1.0, 13 November 2012 http://www.openmobilealliance.org
113	[CONNMO]	OMA Standardized Connectivity Management Objects for use with OMA Device Management, Approved Version 1.0, 7 November 2008 http://www.openmobilealliance.org
114	[CONNMOHTTP]	Standardized Connectivity Management Objects HTTP Proxy Parameters for use with OMA Device Management, Approved Version 1.0, 24 October 2008 http://www.openmobilealliance.org
115	[CPM-SYS_DESC]	OMA Converged IP Messaging System Description, Candidate Version 1.0, 12 October 2010 http://www.openmobilealliance.org
116	[OMA-DM]	OMA Device Management, Approved Version 1.2.1, 17 June 2008 http://www.openmobilealliance.org
117	[DMPRO]	OMA Device Management Protocol, Approved Version 1.2.1, 17 June 2008 http://www.openmobilealliance.org
118	[OMA-EVVM]	Enhanced Visual Voice Mail Architecture and technical Specification Version 1.0, 7 August 2012. http://openmobilealliance.org
119	[MMSCTR]	OMA Multimedia Messaging Service Client Transactions, Approved Version 1.3, 13 September 2011 http://www.openmobilealliance.org
120	[MMSENC]	OMA Multimedia Messaging Service – Encapsulation Protocol, Approved Version 1.3 – 13 Sep 2011 http://www.openmobilealliance.org

Ref	Document Number	Title
121	[MMSMO]	OMA Management Object for MMS, Candidate Version 1.3, 28 January 2008 http://www.openmobilealliance.org
122	[Location_API]	RESTful bindings for Parlay X Web Services – Terminal Location, Candidate Version 1.1, 11 January 2011 http://www.openmobilealliance.org
123	[Presence]	OMA Presence SIMPLE Specification, 1.1, http://www.openmobilealliance.org/
124	[Presence2.0_DDS]	Presence SIMPLE Data Specification, Approved Version 2.0, 29 September 2009 http://www.openmobilealliance.org/
125	[Presence2.1_DDS]	Presence SIMPLE Data Specification, Approved Version 2.1, 2 October 2010 http://www.openmobilealliance.org/
126	[Presence2.0_TS]	Presence SIMPLE Specification, Approved Version 2.0, 10 July 2012 http://www.openmobilealliance.org/
127	[Presence2.0_RLS_TS]	Resource List Server (RLS) Specification, Approved Version 2.0, 10 July 2012 http://www.openmobilealliance.org/
128	[Presence_Content]	Presence Content XDM Specification, Approved Version 2.0, 10 July 2012 http://www.openmobilealliance.org/
129	[PDE_13]	OMA Presence SIMPLE Data Extensions , Approved Version 1.3, 1 November 2011 http://www.openmobilealliance.org/
130	[PRESENCEIG]	Implementation Guidelines for OMA Presence SIMPLE v1.1 Presence http://www.openmobilealliance.org/
131	[PRESENCEMO]	OMA Management Object for Presence SIMPLE, Approved Version 1.0, 25 February 2010 http://www.openmobilealliance.org
132	[PRESENCE2MO]	OMA Management Object for Presence SIMPLE 2.0, Approved Version 2.0, 10 July 2012 http://www.openmobilealliance.org
133	[PresenceXDM]	Presence XDM Specification, Approved Version 1.1, 27 June 2008 http://www.openmobilealliance.org/
134	[REST RCS API]	OMA RCS Profile of RESTful Network APIs, Draft Version 3.0, 31 January 2014 http://member.openmobilealliance.org/ftp/Public_documents/ARCH/RCS-Profile3/Permanent_documents/OMA-TS-REST_NetAPI_RCSProfile_-V3_0-20140131-D.zip NOTE: Reference to be updated with the candidate version once available.

Ref	Document Number	Title
135	[REST WEBRTC SIG API]	OMA RESTful Network API for WebRTC Signalling , Candidate Version 1.0, 11 February 2014 http://member.openmobilealliance.org/ftp/Public_documents/ARCH/Permanent_documents/OMA-TS-REST_NetAPI_WebRTCSignaling-V1_0-20140211-C.zip
136	[RLSXDM]	Resource List Server (RLS) XDM Specification, Approved Version 1.1, 27 June 2008, http://www.openmobilealliance.org/
137	[SHARED-XDM]	Shared XDM Specification, Approved Version 1.1, 27 June 2008 http://www.openmobilealliance.org/
138	[SUPLMO]	OMA Management Object for SUPL, Candidate Version 2.0 – 27 January 2011 http://www.openmobilealliance.org/
139	[XDM1.1_AD]	XML Document Management Architecture, Approved Version 1.1, 27 June 2008 http://www.openmobilealliance.org/
140	[XDM2.0_AD]	XML Document Management Architecture, Candidate Version 2.0, 16 September 2008 http://www.openmobilealliance.org/
141	[XDM1.1_Core]	XML Document Management (XDM) Specification, Approved Version 1.1, 27 June 2008 http://www.openmobilealliance.org/
142	[XDM2.0_Core]	XML Document Management (XDM) Specification, Candidate Version 2.0, 16 September 2008 http://www.openmobilealliance.org/
143	[XDM2.2_Core]	XML Document Management (XDM) Specification, Draft Version 2.2, 24 June 2014 http://www.openmobilealliance.org/
144	[XDMIG]	Implementation Guidelines for OMA XDM v1.1, http://www.openmobilealliance.org/
145	[XDMMO]	OMA Management Object for XML Document Management 1.1, http://www.openmobilealliance.org
146	[vCard21]	vCard, The Electronic Business Card, A versit Consortium Specification, 18 September 1996 http://www.imc.org/pdi/vcard-21.doc
147	[ISO8601]	ISO 8601:2004 Data elements and interchange formats -- Information interchange -- Representation of dates and times, 18 March 2008 http://www.iso.org
148	[W3C WebRTC]	WebRTC 1.0: Real-time Communication Between Browsers http://www.w3.org/TR/webrtc/
149	[W3C WS]	The WebSocket API http://www.w3.org/TR/websockets/

Ref	Document Number	Title
150	[W3C XHR]	XMLHttpRequest http://www.w3.org/TR/XMLHttpRequest/
151	[POSIX]	IEEE Standard for Information Technology—Portable Operating System Interface IEEE Std 1003.1, 2013 Edition

1.5 Differences to previous specifications

RCS 6.0 evolves on the functionality defined for RCS 5.3.

The structure of the document has been modified compared to the previous releases. A section 4 has been added to cover the cross-service functionalities. All procedures related to the Common Message Store (including Common File Store) have been moved to this new section (in sub section 4.1).

The following sub-sections list the major differences.

1.5.1 New features and procedures

- Visual Voice Mail
  - New section (3.13) dedicated to the management of the voicemail messages from the RCS client
- Configuration
  - New Hyper-Text Transfer Protocol (HTTP) parameter to indicate if the RCS messaging client is selected as the default voice mail application (see section 2.3.3.2)
  - New Management Object (MO) Parameters to enable/disable/manage the new services (see Annex A)
  - New MO Parameter to indicate whether the synchronization with the Common Message Store Should remain available in case the cellular data is switched off (see section A.1.15)
- Addressing and identities
  - New caller and callee procedures to match identity when in-call for incoming (2.5.2.2) and outgoing (2.5.3.2) SIP requests
  - New procedure for addressing in a request for services enriching a call (see section 2.5.3.2.1)
- Access technologies
  - Addition of Evolved Packet Core (EPC)-integrated WiFi when sharing the registration with Voice over Wi-Fi (VoWiFi) as defined in [PRD-IR.51]

- Messaging
  - Delivery of sent and received IP Messages to all online devices (see [RCS-CPM-CONVFUNC-ENDORS])
  - Indication that subscriber became capable to receive Short Message Service (SMS) as trigger for deferred delivery of messages (see [RCS-3GPP-SMSIW-ENDORS], [RCS-CPM-IW-ENDORS] and [RCS-CPM-CONVFUNC-ENDORS])
- Group Chat
  - Handling of de-provisioned Group Chat participant (3.4.4.1.3.1 and [RCS-CPM-CONVFUNC-ENDORS])
  - Allow Group Chat initiator to remove participant from Group Chat (see [RCS-CPM-CONVFUNC-ENDORS])
  - Management of ‘subject’ and ‘icon’ in Group Chat (see [RCS-CPM-CONVFUNC-ENDORS])
- File Transfer
  - File transfer fallback (3.5.4.8.6) - procedure to send files and multimedia content to a contact that does not support any of the enabled File Transfer mechanisms
- Content Share
  - Call Composer pre-call service (see sections 3.6.1.3.1, 3.6.4.3 and [PRD-RCC.20])
  - Post-call service for unanswered calls (see sections 3.6.1.3.2, 3.6.4.4 and [PRD-RCC.20])
  - In-call Shared Map and Shared Sketch services (see sections 3.6.1.2, 3.6.4.5, 3.6.4.6 and [PRD-RCC.20])

1.5.2 Removed features and procedures

- Devices and client types
  - Removed Device modes
- Access technologies
  - Removed Co-existence of RCS with Voice over High Speed Packet Access (VoHSPA)
- Access Point Name (APN)
  - Removed RCS-E ONLY APN
- Multi-device
  - No more need for the disposition notifications to target the device that has sent the related message.

- Secondary devices
  - Removed control of service delivery feature
  - Removed the possibility to configure the device to send SMS/MMS messages to the SMS-C/MMS-C without interworking
- Security and Privacy
  - Removal of support of GPRS IMS Bundled Access (GIBA)
- Content Share
  - Removed Video Share outside of a call
- Geolocation
  - Removed support of Secure User Plane Location (SUPL)
- IP Video Call
  - Removed ‘Replace a CS voice with an RCS IP Video call’ feature (section 3.9)
- Configuration parameters
  - Removal of AVAILABILITY AUTHORISATION, FAVOURITE LINK CONTROL, FAVOURITE LINK URLS, FAVOURITE LINK LABEL MAX LENGTH, ICON MAX SIZE, LOCATION TEXT MAX LENGTH, MAX LOCATION UPDATE, PUBLISH EXPIRY TIME, XDM CHANGES SUBSCRIPTION, IM WARN SF, IM WARN IW, IM SMS FALLBACK AUTH, FIRST MSG IN INVITE, MAX CONCURRENT SESSIONS, DISABLE DIRECTION HEADER, FT THUMB, FT STANDFWD ENABLED, tel or SIP URI – international, tel or SIP URI - for non- international format, RCS-E ONLY APN, ENABLE RCS-E SWITCH, ALWAYS USE IMS APN, RCS IP VOICE CALL BREAK OUT, RCS IP VIDEO CALL UPGRADE FROM CS, RCS IP VIDEO CALL UPGRADE ATTEMPT EARLY, RCS IP VIDEO CALL UPGRADE ALLOWED ON CAPABILITY ERROR, IP VIDEO CALL DEFAULT MECH, MAX RRAM DURATION, RCS IP VIDEO DATA OFF

1.5.3 Modified features and procedures

Area	Section	Differences from RCS 5.3
Configuration	[PRD-RCC.14]	Token characteristic now has undefined validity and can be included in all responses (i.e. also when configuring over cellular)
IMS Registration	2.4	Now refers to [PRD-NG.102] for Voice over LTE (VoLTE)/VoWiFi enabled devices.
IMS Registration	2.4.2	Globally Routable User agent URI (GRUU) support made optional for devices

Area	Section	Differences from RCS 5.3
IMS Registration	2.4.3	Simplification of the telephony tag: • it is used to indicate if the device can receive SMSs associated with the identity used for RCS when the device is not registered in IMS for messaging • 'volte', 'vohspa' and 'vowifi' values are removed
IMS Registration	2.4.6	Proxy Call Session Control Function (P-CSCF) discovery refers to [PRD-NG.102] for VoLTE/VoWiFi enabled devices
Capability discovery	2.6.1	Possible disabling by an operator of end to end capability discovery – Interoperability is provided by network based interworking
Capability discovery	2.6.1.1.2, 2.6.1.2.5.1, 2.6.1.3.1	Removal of tags related to Group Chat S&F and VSh outside of a voice call capabilities. FT thumbnail is now only for backwards compatibility purposes. New capabilities for Enriched calling (Call Composer, Post-Call, Shared map, Shared sketch)
Image orientation	2.7.1, 3.6.4.1.2, 3.6.4.1.3	Removal of image orientation extension for video section. The procedures are now standardized in 3GPP and referred to there
File Transfer over HTTP	3.5.4.8.3	Recommended support for Single Sign On (SSO, use of a cookie) between the client and Content Server.
Audio Messaging	3.11.4.2.1	The maximum duration Audio Message is not configurable anymore, it has been set to a hard limit of 10 minutes. Removal of the MAX RRAM DURATION configuration parameter.
Message Store	4.1, B.4, [RCS-CPM-MSGSTOR-ENDORS]	Modification of the message archiving solution. Introduction of the Archived flag. Removal of the RCSMessageStore folder
Message Store	4.1.4.4	Update of the Message-Correlator algorithm
Message Store	4.1.6.5, 4.1.6.6, 4.1.6.8	Clarification on the client procedures for setting the \Seen and \Deleted flags and synchronizing.
Message Store	4.1.6.8, B.4, [RCS-CPM-MSGSTOR-ENDORS]	Add support of XLIST-CHANGEDSINCE
Message Store	[RCS-CPM-MSGSTOR-ENDORS]	Update of the Group State Object
Event Reporting Framework	4.1.4.8, [RCS-CPM-CONVFUNC-ENDORS]	Follows the procedures of OMA CPM 2.1
Configuration parameters	A.1.4.3	MAX SIZE 1-to-1 IM becomes MAX SIZE IM
Configuration parameters	A.1.4.3	Possibility to configure Standalone messaging for reception only

Area	Section	Differences from RCS 5.3
Configuration parameters	A.1.7.1	Addition of Table 78 presenting the Usage of [3GPP TS 24.167] MO parameters for RCS
Configuration parameters	A.1.10	CAPABILITY DISCOVERY MECHANISM becomes optional and gains a new value (2 : to disable the capabililty discovery mechanism)
Configuration parameters	A.1.15	Evolution of IR94 DATA OFF that becomes IP VIDEO CALL DATA OFF

Table 1: Modifications from RCS 5.3

2 RCS General Procedures

2.1 RCS architecture

For RCS, the one mandatory network element is the IMS core system which enables peer-to-peer communication between RCS clients. Other network nodes can be deployed by the Service Provider to provide additional parts of the RCS feature set. Figure 3 illustrates a simplified example of the RCS architecture; a Service Provider may choose a different approach to implement a function within the Service Provider domain not influencing the interoperable NNI aspects.

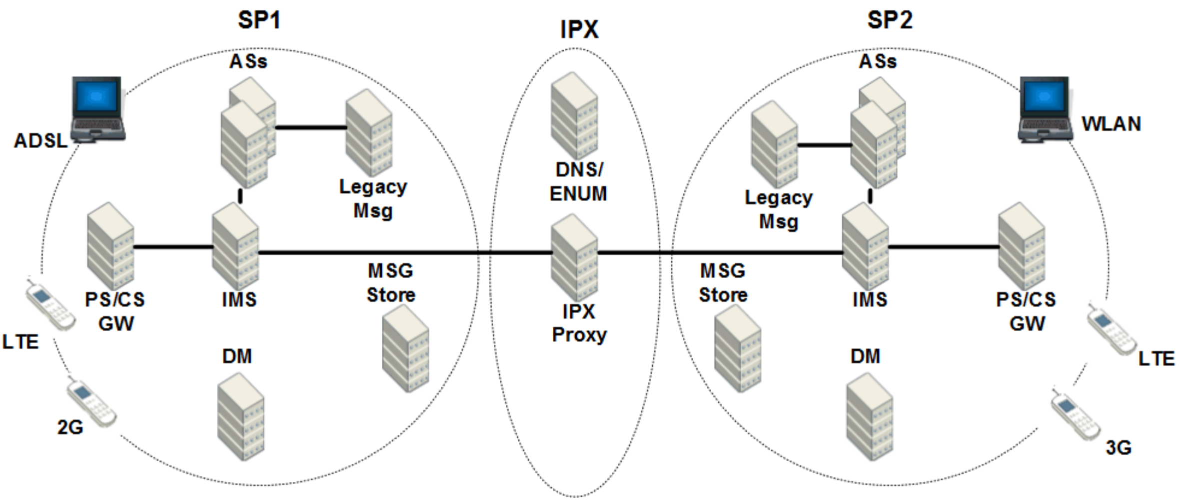

Figure 3: Simplified Example of RCS Architecture

The PS/CS gateway (GW) is used for interworking between Circuit Switched (CS) and Packet Switched (PS) voice, for example, Voice over Long Term Evolution (VoLTE). Msg Store relates to the CPM (Converged IP Messaging) Message Store Server as illustrated in section 3.2. Legacy Msg refers to the Short Message Service (SMS)/Multimedia Message Service (MMS) services that may be utilized via an IWF (Interworking Function) located in the group of Application Servers (ASs) which in addition to these IWF node(s) may also include various other nodes used by the RCS services, for example:

- Presence Server
- Messaging Server
- XML (Extensible Markup Language) Document Management (XDM) Server (XDMS)
- Multimedia Telephony (MMTEL) Application Server

Figure 3 shows examples of two RCS Service Providers exchanging traffic with each other using the standard Network-to-Network Interface (NNI) mechanisms (IPX, IP Packet Exchange) as documented in [PRD-IR.90].

RCS compliant access networks include, but are not limited to, those illustrated in the Figure 3. Thus, deploying the RCS service does not indicate a 3G network should always be deployed. Further details of RCS services relating to particular access networks are found in section 2.7. It is recommended to follow the IMS node naming guidelines as defined in [PRD-IR.67] for naming the Messaging Server, since this server is required to be addressable across IPX.

2.2 RCS devices and client types

RCS defines two types of devices:

1. **Primary device**: a device carrying a SIM that is associated with the identity (i.e. IMPU/MSISDN) used for RCS. Two types of RCS clients exist for such a device that connect directly to the IMS:

    a) **RCS embedded client**: This is the client that is provided as part of the handset implementation and it is fully integrated with the native applications (address book, gallery/file browser application, calling application, etc.). This type of RCS client shall represent the identity of the device as per [PRD-NG.102] when enabled for VoLTE and VoWiFi. Otherwise section 2.4.2 and [3GPP TS 24.229] apply and the International Mobile Station Equipment Identity (IMEI) shall be used in sip.instance during registration.

    b) **RCS downloadable client**: This is a client providing its own IMS connectivity that may be preinstalled or that has to be downloaded by the user. However it is not part of the device base software (i.e. it has no access to internal Application Programming Interfaces [APIs] and advanced Operating System [OS] functionality). The level of integration with the native applications is limited to the possibilities permitted by the corresponding mobile OS or OS platform API. Consequently, the RCS client shall represent the identity of the device as per section 2.4.2, but, the IMEI shall not be used in sip.instance during registration.

NOTE1: Next to this, there may also be downloadable clients that use terminal APIs (e.g. [PRD-RCC.53]) to access the RCS functionality that is provided by a device's RCS embedded client. This type of client is not considered in this document because it does not alter the UNI which is handled by the device's RCS embedded client.

2. **Secondary device**: a device that does not carry a SIM that is associated to the identity used for RCS

NOTE2: It may happen that a secondary device carries a SIM (e.g. a tablet or PC providing cellular data connectivity). That SIM will be associated to a different identity than the one used for RCS though.

RCS services can also be deployed using an identity that is not linked to a mobile network (e.g. a fixed-line telephone number or other identity). [PRD-RCC.14] describes in section

2.6 a generic mechanism for the configuration of such clients. Because it will be dependent on the use case, it is out of scope of this document how an identity is assigned to such devices and on which basis they are considered as a primary or secondary device.

2.3 Configuration Procedures

A user can only initiate the use of RCS services once their client is configured and the corresponding subscriber (uniquely identified by the relevant IMS Unique Resource Identifier [URI]; that is a tel URI and/or a SIP URI) is provisioned by the RCS Service Provider to access the RCS services.

Both processes should be performed automatically (e.g. when a subscriber first turns on their RCS capable device and connects with their Service Provider). This gives the end user the impression that the RCS services are working out of the box and minimises operational impacts to Service Providers.

2.3.1 First-time Start of an RCS capable device

A mobile network offering RCS services to its subscriber base should be able to detect when a user connects to the network for the first time with an RCS capable device. Upon detecting a user connection, two processes are triggered to execute:

1. Service provisioning: the process whereby the relevant configuration is performed on the network elements to make the RCS services available to the user (e.g. provisioning an account on the IMS core and relevant application servers).

NOTE1: In addition to this auto-provisioning on first usage, the service may be provisioned in advance by the Service Provider.

2. Client configuration: the process whereby the network provides the client with its configuration using one of the mechanisms described in section 2.3.3.

As shown in the Figure 4, an RCS capable device must successfully complete service provisioning and configuration procedures before it can be used. The service provisioning and configuration procedure may be triggered in a variety of different ways, including

- An RCS capable device is powered on: as a result, the network may be able to identify or detect that the user/device pair can use RCS services and, as a consequence, trigger the relevant device configuration procedures described in more detail later in this section.

NOTE2: The triggering process is network specific and outside the scope of this specification.

- The RCS capable device may also be able to perform a customised bootstrap operation (also, named factory bootstrap) to trigger a client-initiated Open Mobile Alliance Device Management (OMA-DM) session towards an OMA-DM server for client configuration purposes.

An alternative to this automated mechanism could be a manually triggered configuration (e.g. by a menu item or requested by an operator in a store).

If not configured, the default behaviour for RCS shall be to not offer the RCS services and to disable the RCS specific service entry points.

2.3.1.1 First-time use scenario

The assumption in this scenario is that User A is entitled to access RCS services (because for example User A's tariff includes the ability to use the RCS services) however User A has never used an RCS-capable device before.

Prior to the first-time registration, it is necessary to provision the user on the network (e.g. by auto-provisioning) and to configure the user's RCS client with the correct settings as described further in this section 2.3. When the provisioning and client configuration is completed, the first-time registration procedure can take place.

RCS first-time registration consists of the following:

1. Register (as described in section 2.4)
2. Establish (i.e. find) a subset among User A's existing contacts (if any) who are also RCS users (as described in section 2.6.2).

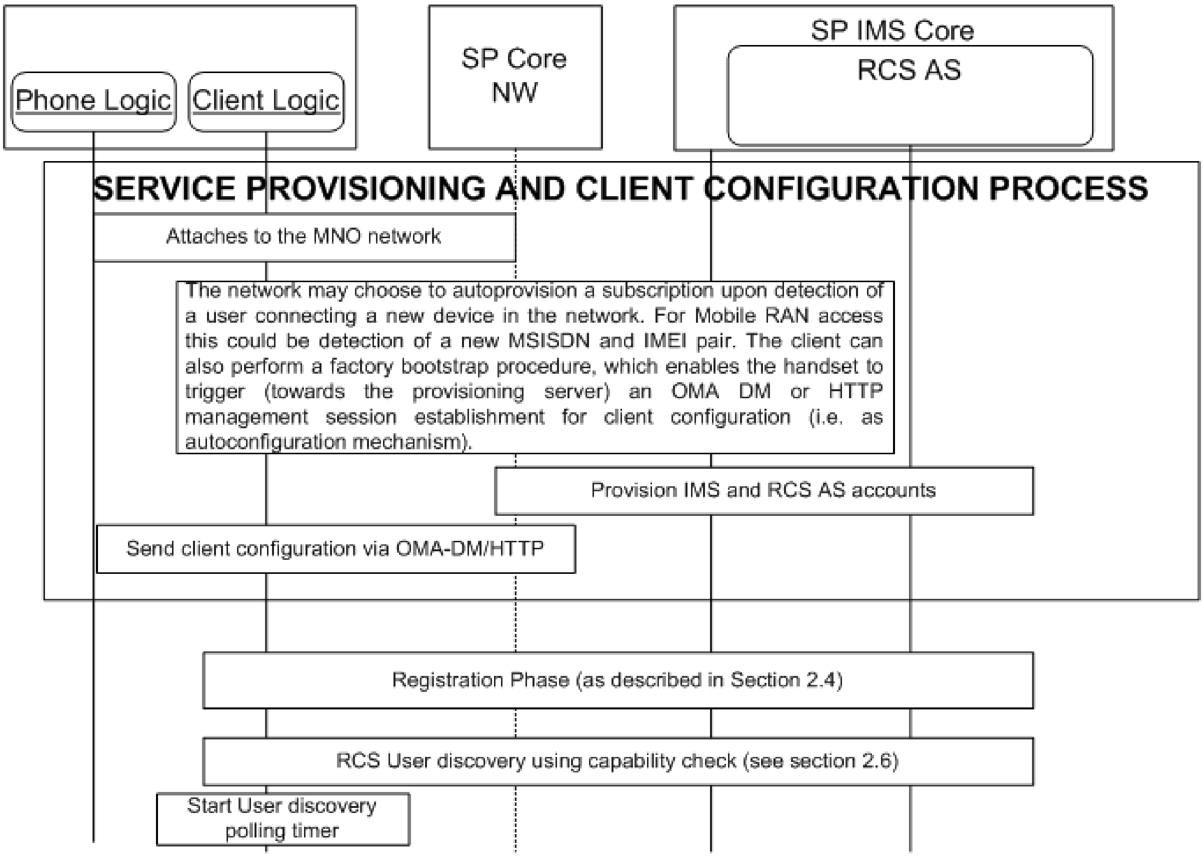

Figure 4: First-Time Start of an RCS capable device - Sequence Diagram

2.3.1.2 Additional first-time configuration scenarios

In addition to the scenario described in the section 2.3.1.1 (first time the user registers with the IMS network), there are several additional scenarios where the same sequence applies:

- When the user changes to another RCS enabled device: In this scenario, the sequence is identical to that described in section 2.3.1.1; however, the IMS provisioning process (i.e. provisioning IMS and RCS AS accounts) is not required as it has already been performed for this subscriber.
- When a customer changes the Subscriber Identity Module (SIM) card in the device: In this scenario, the sequence is identical to the one described in the section 2.3.1.1.
- A configuration update is required that implies changes in the user's IMS identity (that is a different tel URI and/or SIP URI). In this scenario, the sequence is identical to the one described in the section 2.3.1.1.
- A configuration update is required that implies the use of a different capability discovery mechanism: As described in section 2.6, switching the capability discovery mechanism parameter automatically triggers the RCS first-time registration process. This parameter is described in Annex A (section A.1.10).

2.3.2 Client configuration parameters

The set of client configuration settings is presented in Annex A Managed objects and configuration parameters.

All the RCS client configuration parameters must be restricted from being modified by the user.

In a device enabled for VoLTE or VoWiFi that is configured to share a registration between RCS and Multimedia Telephony (see the configuration parameter RCS VOLTE SINGLE REGISTRATION defined in section A.1.7), the default IMS settings as defined in [PRD-IR.92]/[PRD-IR.51] are used on cellular and EPC-integrated Wi-Fi access respectively. Therefore, as stated in section 3.1.1.1 of [PRD-RCC.15], some client configuration parameters referred to in section 3.1.1.2 of [PRD-RCC.15] are not used in this case.

NOTE: Secondary devices (see section 2.2) cannot be enabled for VoLTE or VoWiFi. Similarly downloadable clients (see section 2.2) do not have sufficient device integration to behave as a device that is enabled for VoLTE or VoWiFi.

Following a successful configuration, the provided settings are active/updated and used on RCS client and the RCS client is ready to register with the network. Once this registration process has successfully completed, the user is able to make use of the RCS services.

These client configuration parameters could also be updated later by the Service Provider by pushing new configuration documents using the OMA-DM enabler or the HTTP configuration mechanisms defined in section 2.3.3.

2.3.3 RCS client autoconfiguration mechanisms

2.3.3.1 Overview

This specification provides two mechanisms that can be used to perform the autoconfiguration of the configuration parameters controlling the RCS functionality in terminals carrying the SIM associated with an RCS user's main identity:

1. [OMA-DM]: This is based on the managed object configuration outlined in Annex A, section A.2. If the RCS capable device supports [OMA-DM], then the following requirements shall be supported by the device:

    - Multiple management authorities in which the Service Provider Device Management accounts are persistent, non-editable and non-visible to an RCS user (e.g. Software (SW) updates do not delete/overwrite DM accounts) and are only accessible to the respective active Service Provider DM account (protected by the OMA-DM Access Control List (ACL) mechanism).
    - The active Service Provider's DM account should be selected and activated after a SIM card change.
    - The RCS configuration parameters are protected against non-Service Provider authorities (through the OMA-DM ACL mechanism).
    - Each RCS Service Provider should have its own RCS Management Object (MO) sub-tree and the OMA-DM account shall have access to the device settings (e.g. for the purpose of access settings configuration).
    - The 'active' Service Provider's RCS MO sub-tree needs to be visible to that Service Provide (i.e. not to the end-user), selected and activated after a SIM card change.
    - An RCS capable device shall support the customised bootstrap procedure (also, named factory bootstrap).That means the Service Provider OMA-DM account, including OMA-DM server address, is pre-loaded into the device at factory phase. This procedure is specified in section 5.1.2.1 of OMA Device Management Bootstrap (see [OMA-DM]). The customised bootstrap triggers a client-initiated management session from the RCS capable device towards a pre-provisioned OMA-DM server (operated by the Service Provider which the device is subscribing to, i.e. the Home Public Land Mobile Network (HPLMN)). This provides an OMA-DM client with the possibility to initiate and perform the RCS autoconfiguration procedure.
    - The scenarios under which a device shall perform the factory bootstrap procedure[1], are as follows:

        - When a device is switched on for the first time;
        - When a factory reset was performed;
        - When a user changes the SIM card on the device and there is no stored configuration for the new SIM;
        - After a device software update is applied which introduces or extends the RCS client functionality in the terminal

    - After successfully processing the customised bootstrap procedure, the DM client of an RCS capable device shall automatically initiate a management session to the DM server configured in the bootstrap at the next practical opportunity[2] (that

[1] The Service Provider network may trigger an OMA-DM configuration when a configured SIM is used in a different device.

[2] This is an additional requirement compared to the OMA DM 1.2.1 specifications.

is when network connectivity and other factors would allow such a connection to occur).

2. Another client configuration mechanism defined in [PRD-RCC.14] including the enhancements described in section 2.1 of [PRD-RCC.15] of which the RCS use is described in section 2.3.3.2, is based on the (Secure) Hypertext Transfer Protocol (HTTP/HTTPS) and has the following main goals:

   - Enabling a configuration procedure that is transparent to the RCS user
   - Reducing the complexity of the auto-detection mechanism on the network infrastructure

For the configuration of additional RCS capable devices (i.e. devices not carrying the SIM associated with a subscriber’s main identity), the HTTP(S) mechanism shall be used as described in [PRD-RCC.14] and [PRD-RCC.15] for such devices.

NOTE: Although RCS provides different mechanisms to perform the auto-configuration, the configured parameters remain the same and are independent of the mechanism that is used. The used mechanism only determines the used protocol and the encoding of the parameters between the client and the network.

2.3.3.2 RCS use of HTTP(S) based client configuration

An RCS device supporting HTTP configuration as described in [PRD-RCC.14] shall support all enhancements described in section 2.1 of [PRD-RCC.15] including those relying on End User Configuration Requests (EUCR) covered in sections 2.1.2 and 2.1.3.1 of [PRD-RCC.15].

Next to the parameters defined for the different generic cases in [PRD-RCC.14], an RCS client shall also include the following information as HTTP GET parameters in the configuration requests:

Parameter	Description	Mandatory	Format
rcs_state	This is either -4, -3, -2, -1, 0 or a positive integer. 0 indicates that the RCS configuration must be updated (e.g. the configuration is damaged; non-existent or an update is needed following a SIM change). A positive value indicates the version of the configuration document in which the last RCS configuration was received i.e. in most cases this will match the value of the generic *vers*parameter defined in [PRD-RCC.14]. -1 indicates that the operator has disabled the RCS services on the device/client (i.e. the last configuration document included a RCS DISABLED STATE configuration parameter set to -1). -2 Indicates that RCS is disabled on the device, but a configuration query might be triggered on user action (i.e. the last configuration document included a RCS DISABLED STATE configuration parameter set to -2). -3 indicates that RCS is in a dormant state (i.e. no registration) in a way that is transparent to the user (i.e. the last configuration document included a RCS DISABLED STATE configuration parameter set to -3). -4 indicates that the user has explicitly disabled the RCS services on the device/client.	Y	Int (-4,-3, -2, -1, 0 or a positive integer)
rcs_version	String that identifies the RCS version supported by the client. For this release it shall be set to “6.0” (without the quotes)	Y	String (4 max), Case-Sensitive
rcs_profile	String that identifies a fixed set of RCS services that are supported by the client. The services that are supported and the value to be used for the rcs_profile parameter to reference to this set are to be defined in external documents (e.g. a Service Provider’s RCS Service definition document). In case multiple, (potentially overlapping) sets are supported the parameter shall be included multiple times	N	String (15 max), Case-Sensitive
client_vendor	String that identifies the vendor providing the RCS client.	Y	String (4 max), Case-Sensitive

Parameter	Description	Mandatory	Format
client_version	String that identifies the RCS client version. client_version_value = Platform "-" VersionMajor "." VersionMinor Platform = Alphanumeric (9 max) VersionMajor = Number (2 char max) VersionMinor = Number (2 char max) Example: client_version=RCSAndrd-1.0	Y	String (15 max), Case-Sensitive
default_sms_app	This is either 0,1 or 2 0 indicates that the OS does not allow user to select SMS application or the client cannot identify the selected SMS application 1 indicates that the RCS messaging client is selected as the default SMS application 2 indicates that the RCS messaging client is not selected as the default SMS application	N, only mandatory for client platforms supporting SMS (i.e. primary devices)	Int (0,1,2)
default_vvm_app	This is either 0 or 1 0 or absent indicates that the RCS messaging client is not selected as the default VM application. 1 indicates that the RCS messaging client is selected as the default VM application	N, only mandatory for client platforms supporting VVM client (i.e. primary devices)	Int (0,1)

Table 2: HTTP configuration: additional RCS specific HTTPS request GET parameters

When the RCS client is enabled (i.e. last received configuration did not include the RCS DISABLED STATE configuration parameter, see section A.1.1), a change by the user of the selected SMS application that would result in a different value of the default_sms_app parameter shall also trigger a configuration query.

When the RCS client is enabled (i.e. last received configuration did not include the RCS DISABLED STATE configuration parameter, see section A.1.1), a change by the user of the selected VVM application that would result in a different value of the default_vvm_app parameter shall also trigger a configuration query.

2.3.3.2.1 RCS client management

During configuration the service provider may want to control the general state of the RCS client (e.g. disable it) without affecting the other services configured through the same configuration document. This shall be done through the generic RCS DISABLED STATE client configuration parameter defined in section A.1.1. Only if not included, other RCS settings shall be included in the configuration document. The client shall ignore any such settings if included when also the RCS DISABLED STATE parameter is included in the document.

If the Service Provider chooses to temporary disable RCS functionality on the device/client, the RCS DISABLED STATE configuration parameter shall be set to 0.

If the Service Provider chooses to permanently disable the RCS functionality on an RCS capable device/client, the RCS DISABLED STATE configuration parameter shall be set to -1.

If the SIM is swapped or the device is reset, the RCS capable device shall again set the rcs_state parameter defined in Table 2 to 0 in the queries for configuration settings.

If the Service Provider chooses to disable the RCS functionality on an RCS capable device/client until there is a UI dependent user action triggering a new query, the RCS DISABLED STATE parameter shall be set to -2.

If the SIM is swapped or the device is reset or the user triggers the UI dependent action, the RCS capable device shall initiate a configuration request including the rcs_state parameter set to 0.

If the Service Provider chooses to put the RCS functionality on an RCS capable device/client in a dormant state, the RCS DISABLED STATE parameter shall be set to -3.

The RCS client shall after receiving such a response behave as follows:

- It shall perform the configuration queries as if it were configured with a valid document (e.g. it performs a query at reboot, when an SMS requesting reconfiguration is received, etc.). In those queries it shall provide as value for the rcs_state parameter ‘-3’.
- The existing configuration document remains valid (i.e. a response with the RCS DISABLED STATE parameter set to ‘-3’ shall be handled in this aspect as if the rcs_state parameter matched the current version available in the client).
- It shall not register into the IMS until a subsequent configuration query results in a configuration XML that does not contain the RCS DISABLED STATE configuration parameter. If it was registered when the document with the RCS DISABLED STATE configuration parameter set to ‘-3’ is received, the client shall unregister.
- All RCS services entry points shall remain available (including those that base on cached capabilities). When the user activates RCS through one of those capabilities, the client shall:

  - Perform a configuration query providing as value for the version parameter the version of the latest configuration document that was received by the client (i.e. a positive value) including the rcs_state parameter with as value the version of the last configuration document that included a full RCS configuration (i.e. not including the RCS DISABLED STATE parameter).
  - If a new configuration document is received or the previously received document is still valid, apply that document, register into the IMS and perform a capability query to verify that the requested action is possible. If not the action is not possible, inform the user of this situation. Keep RCS active afterwards.
  - If an error or a document with a negative version is returned, inform the user that RCS is not available at that time.

- Provide an indication to the user during these actions to show that RCS is being activated

- These actions to activate RCS shall also be performed when a SMS message requesting reconfiguration is received.

2.4 IMS registration

2.4.1 General

Prior to the registration, the device must be configured as described in section 2.3.

The device and IMS core network must follow the SIP registration procedures defined in [3GPP TS 24.229], complemented with the modifications described in this document (e.g. non registration of some feature tags).

NOTE: In particular, the device shall support the nonce storing procedures as defined in [3GPP TS 24.229-rel12] section 5.1 to allow some traffic reduction.

If the device is enabled for VoLTE or VoWiFi then it must additionally follow the procedures for registration specified in [PRD-NG.102].

In all cases, the device shall register in IMS indicating whether the device can receive SMSs associated with the identity used for RCS when the device is not registered in IMS for messaging using the telephony feature tag described in section 2.4.3.

If the device is not enabled for VoLTE or VoWiFi as defined in section 2.9.1) and it is configured to support RCS IP Voice Call and/or RCS IP Video Call, it shall always register in IMS taking into account the rules defined in respectively Table 50 and Table 51 in sections 3.8 and 3.9.

NOTE: Secondary devices (see section 2.2) cannot be enabled for VoLTE or VoWiFi. Similarly downloadable clients (see section 2.2) do not have sufficient device integration to behave as a device that is enabled for VoLTE or VoWiFi and will, therefore, have to assume that the device is not enabled for VoLTE or VoWiFi.

When the domain selection has selected IMS voice and the device is configured to share a registration between RCS and Multimedia Telephony (see the configuration parameter RCS VOLTE SINGLE REGISTRATION defined in section A.1.7), the device is using VoLTE or VoWiFi, and shall not directly register in non-cellular networks (i.e. it shall not directly register over a Wi-Fi network). The client may register over EPC integrated Wi-Fi according to [PRD-NG.102].

Otherwise, the device may de-register from IMS on the cellular network and register again through direct non-cellular access when that is available. This switch to direct non-cellular access will interrupt any ongoing RCS sessions.

When registered over direct non-cellular access, all RCS traffic including the traffic from supporting protocols (i.e. XCAP, HTTP and IMAP) shall use this direct non-cellular connection.

As soon as the domain selection is again using IMS voice and the device is configured to share a registration between RCS and Multimedia Telephony (see the configuration parameter RCS VOLTE SINGLE REGISTRATION defined in section A.1.7), the device shall attempt to de-register from IMS through the non-cellular access and shall register again using IMS over the cellular or EPC-integrated Wi-Fi network access.

For a device that is not configured to use VoLTE or VoWiFi or that is configured to not share a registration between RCS and Multimedia Telephony (see the configuration parameter RCS VOLTE SINGLE REGISTRATION defined in section A.1.7), the client sends a SIP REGISTER message to the network using the configuration parameters (SIP proxy and other IMS parameters as presented in section 2.2.1 of [PRD-RCC.15]).

The device must use the authentication mechanisms as described in section 2.13.

If the registration is not successful, the user should not be able to access any RCS services and all RCS contacts services/capabilities shall be reported to the user as not available independently of any setting (the IM CAP ALWAYS ON setting presented in Table 75 is ignored for example). When it is the device's network status that prevents the client from registering (e.g. no PS or Wi-Fi connectivity because the device is in "airplane mode") and the IM CAP ALWAYS ON setting is enabled, the chat service may be shown as available even if the client is not registered.

Finally, note that a precondition to register is that all of the mandatory parameters presented in section A.1 are correctly configured. This includes those optional parameters that, due to their dependency on the configured value of a mandatory parameter, have become mandatory.

2.4.2 Procedures for multidevice handling: GRUU and sip.instance

The device shall support using sip.instance and may support using Globally Routable User agent URIs (GRUUs) to allow an Application Server to uniquely address each RCS client residing on different devices as specified in [3GPP TS 24.229] taking into account the clarifications given below.

When the user agent generates a REGISTER request (initial or refresh), it shall include the Supported header field in the request. The value of that header field may include "*gruu*" as one of the option tags if supported by the client. This indicates to the registrar for the domain that the User Agent (UA) supports the GRUU mechanism.

In each contact included in the REGISTER request, the client shall include a "*sip.instance*" tag, whose value is the instance ID that identifies the user agent instance being registered. As network support for GRUU is not mandatory, sip.instance can be used instead. An RCS client will use a public GRUU if provided by the network, but there is no requirement in RCS for a device to use a temporary GRUU.

If the device is enabled for VoLTE or VoWiFi, then sip.instance shall be set as specified in [PRD-NG.102].

If the RCS client has access to the device IMEI and the device is not enabled for VoLTE or VoWiFi, then sip.instance shall be the IMEI value as per [3GPP TS 24.229]. Otherwise, the value of sip.instance shall use either:

- The value provided as part of the device/client configuration (uuid_Value, as described in [PRD-RCC.15]) shall be used. In this case, the network shall follow one of the algorithms described in [RFC4122], 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.

If the REGISTER response is a 2xx and the network supports GRUU, each Contact header field will contain a "pub-gruu" conveying the public GRUU for the user agent instance. The GRUU support is not mandatory for the Service Providers. Therefore user agents shall not always expect to receive a GRUU from the registrar.

2.4.2.1 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.

2.4.3 Telephony feature tag

RCS defines a telephony feature tag used to indicate to the IMS network whether the device supports CS telephony services and hence can receive SMSs associated with the identity used for RCS when the device is not registered in IMS for messaging. The feature tag shall be included in the Contact header at registration with possible values to include: "*none*" or "*cs* ". The use of any other value carried in the feature tag is out of scope for this specification. If no value is included in the feature tag, it is treated by the IMS network as if it carried the value of "none".

The feature tag is defined as +g.gsma.rcs.telephony=<values>.

Using the terms primary and secondary devices as defined in section 2.2:

- For a secondary device that by definition does not support CS telephony and thus does not support receiving SMSs for the identity used for RCS, the feature tag shall either not be present at all or be set as follows: +g.gsma.rcs.telephony="none".
- For a primary device that supports CS telephony and thus supports receiving SMSs for the identity used for RCS, the feature tag shall be set as follows: +g.gsma.rcs.telephony="cs".

2.4.4 Services feature tags

2.4.4.1 Service related feature tags at IMS registration as per service specifications endorsed by RCS

The client shall include the feature tags related to the authorised/enabled services in the REGISTER request as per the relevant service specifications (e.g. +g.oma.sip-im as per [RCS-SIMPLEIM-ENDORS]).

For the standalone messaging service, when it is enabled i.e. the configuration parameter STANDALONE MSG AUTH defined in section A.1.4.3 is set to 1 or 2, the client shall include the standalone messaging tags defined in [RCS-CPM-CONVFUNC-ENDORS] in the REGISTER request.

2.4.4.2 File Transfer via HTTP feature tags at IMS registration

When File Transfer via HTTP is enabled (the configuration parameters *FT HTTP CS URI, FT HTTP CS USER* and *FT HTTP CS PWD* defined in section A.1.5 are correctly set), the client shall include the File Transfer via HTTP IARI defined in Table 7 in the REGISTER request.

2.4.4.3 Geolocation PUSH feature tags at IMS registration

When Geolocation PUSH is enabled (see PROVIDE GEOLOCATION PUSH in section A.1.8.2), the client shall include the Geolocation PUSH IARI defined in Table 11 in the REGISTER request.

2.4.4.4 RCS IP Call feature tags

When RCS IP Video Calls are enabled for any cellular access (see PROVIDE RCS IP VIDEO CALL in section A.1.12), an RCS client shall when registering in the IMS over a cellular access include the feature tags for RCS IP Calls according to Table 51.

When RCS IP Voice or Video Calls are enabled for Wi-Fi (see PROVIDE RCS IP VOICE CALL and PROVIDE RCS IP VIDEO CALL in section A.1.12), an RCS client shall when registering in the IMS over Wi-Fi access include the feature tags for RCS IP Calls according to Table 50 or Table 51. There is no need for the device to remove the RCS IP Voice/Video service capability once it has registered with it.

RCS defines the RCS IP call feature tags used to indicate the wanted behaviour as listed in Table 3.

RCS service	Tags
RCS IP Voice Call	+g.gsma.rcs.ipcall; +g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.mmtel"
RCS IP Video Call	+g.gsma.rcs.ipcall; +g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.mmtel";video
RCS IP Video Call where video media cannot be removed by the user	+gsma.rcs.ipcall;+g.gsma.rcs.ipvideocallonly;+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.mmtel";video

Table 3: RCS IP Voice/Video Call feature tags

See Table 50 and Table 51 for the media feature tags required at IMS registration and at session setup.

NOTE: IP Voice/Video Call identification for calls without the RCS IP Call tags is as per [PRD-IR.92]/[PRD-IR.94]. In other words, it is up to a service provider to decide upon call breakout.

2.4.4.5 RCS VV-Mail feature tags

When RCS Visual Voice/Video-Mail (VV-Mail) is enabled (RCS VV-MAIL AUTH configuration parameter defined in section A.1.16), an RCS client shall register with the feature tags for RCS VV-Mail as listed in Table 4.

RCS service	Tags
RCS VV-Mail New Message notification	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.vvm"
RCS multi-device sync	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.mdsync"[3]
RCS VV-Mail and VMS sync	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.vmssync"

Table 4: RCS VV-Mail feature tags

When the RCS client is configured to use RCS VV-Mail service, the client should register with the feature tags *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.vvm"* and *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.mdsync"*. If the device is configured to use the traditional Visual Voice Mail (VVM) client, then the client should not register for the RCS VV-Mail service.

An RCS client by registering with the feature tag *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.vvm"* indicates that it is capable of receiving the new voicemail notification events.

An RCS client by registering with the feature tag *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.mdsync"* indicates that it is capable of receiving the voicemail message status change (like read or delete) and also, that it is capable of generating the voicemail message status change events (e.g. when the messages are read or deleted from the client).

The secondary device(s) shall register with the VMS sync feature tag when the primary device is using the traditional VVM service for the voicemail messaging. When the primary device is using the RCS VV-Mail service, then it is not required for the RCS VV-Mail and VMS to sync. When the secondary device is configured with the RCS VV-MAIL VMS SYNC configuration parameter defined in section A.1.16 set to enabled, then the client should register with the feature tag *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.vmssync"*.

NOTE: A primary device shall not register with the *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.vmssync"* feature tag.

2.4.4.6 Extension feature tags

An Extension is enabled or disabled via operator specific means. For example, an operator may make use of the EUCR mechanism as describe in section 2.10 to enable or disable an Extension. Only enabled Extensions are allowed to use the RCS infrastructure.

If an RCS Client supports Extensions that are allowed to use the RCS infrastructure, then when registering in the IMS the RCS Client shall include

[3] This feature tag is for now used only for VV-Mail, but may in future be used in other multi-device cases as well.

- The RCS Extension to Extension data channel ICSI(s) (as per section 3.12.4.2.2.1) if at least one enabled Extension uses this service, if and only if the configuration parameter ALLOW RCS EXTENSIONS (as defined in section A.1.14) is set to 1.
- The IARIs of the enabled Extensions as defined in Table 17, for the Extensions using the RCS Extension to Extension service or any RCS service, if and only if the configuration parameter ALLOW RCS EXTENSIONS (as defined in section A.1.14) is set to 1.

When an Extension is enabled after initial registration, a re-REGISTER request with the new IARI included in the list of media feature tags carried in the Contact header shall be sent out immediately.

When an Extension is disabled by the operator, the Extension's IARI is removed from the list of media feature tags carried in the Contact header of the SIP REGISTER request.

NOTE: To remove an IARI, the device may wait till the next scheduled refresh re-REGISTER request or may issue a re-REGISTER request immediately.

2.4.4.7 Shared Map and Shared sketch feature tags

When Shared Map service is enabled (the configuration parameter SHARED MAP AUTH defined in section 2.1.2 of [PRD-RCC.20] is set to 1), the client shall include the ICSI pertaining to the Enriched Calling Shared Map defined in Table 14 in the REGISTER request.

When Shared Sketch service is enabled (the configuration parameter SHARED SKETCH AUTH defined in section 2.1.2 of [PRD-RCC.20] is set to 1), the client shall include the ICSI pertaining to the Enriched Calling Shared Sketch defined in Table 15 in the REGISTER request.

2.4.4.8 Call composer and post-call feature tags

When Call composer service is enabled (the configuration parameter COMPOSER AUTH defined in section 2.1.2 of [PRD-RCC.20] is set to 1), the client shall include the ICSI pertaining to the Enriched Calling Call composer defined in Table 12 in the REGISTER request.

When Post-call service is enabled (the configuration parameter POST CALL AUTH defined in section 2.1.2 of [PRD-RCC.20] is set to 1), the client shall include the ICSI pertaining to the Enriched Calling Call composer defined in Table 13 in the REGISTER request.

2.4.5 Registration flows

Prior to the first IMS registration, it is necessary to provision the user on the network (e.g. by auto-provisioning) and to configure the device with the required settings (see chapter 2.3). When the provisioning and device configuration phase is completed, the IMS registration takes place. Figure 5 shows the registration flow which applies for all authentication mechanisms covered in Section 2.13. After successful IMS registration, the device performs a capability and new user discovery procedure as described in Section 2.6.

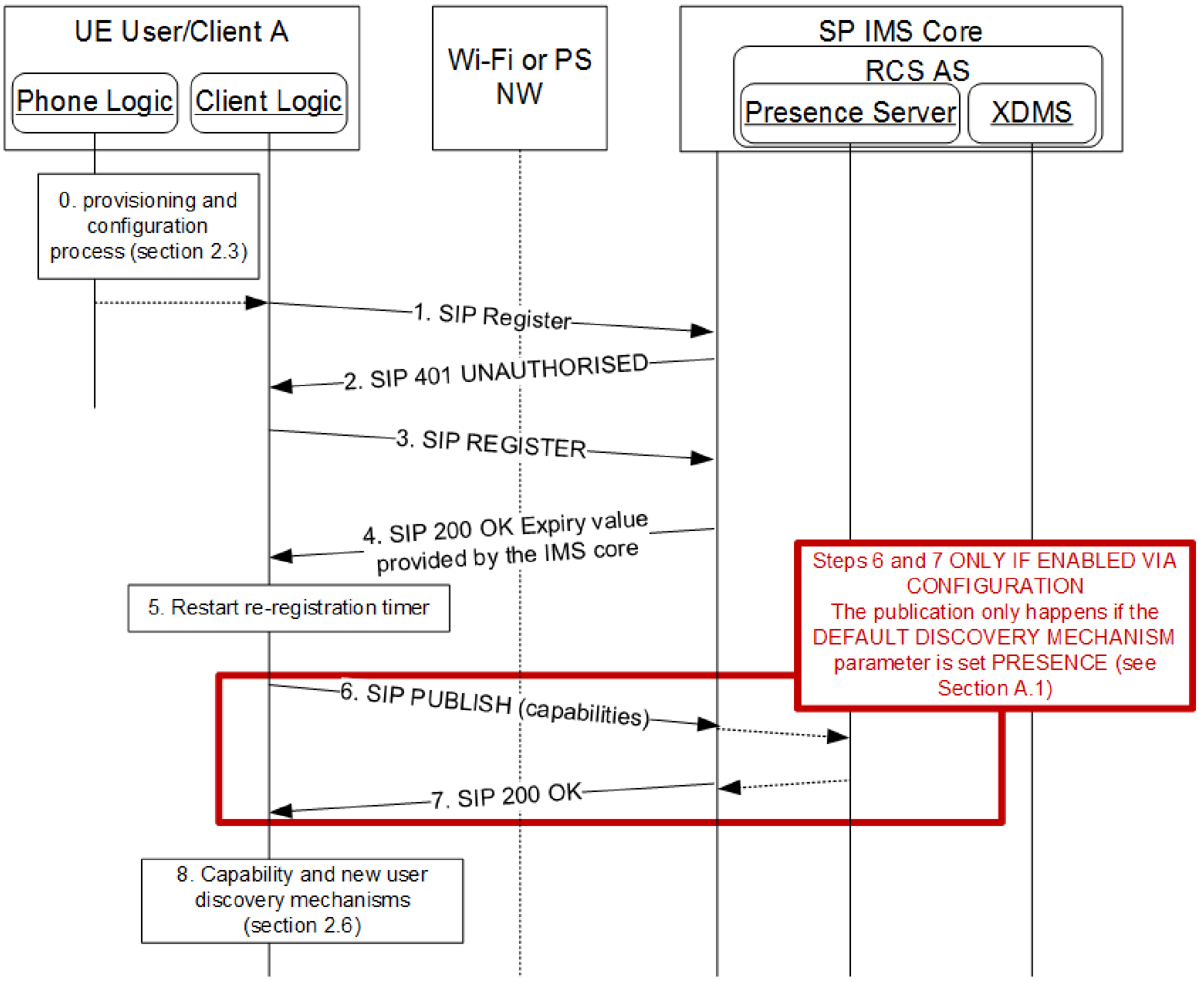

Figure 5: IMS registration flow

Once registered the re-registration timer is initiated which is used to refresh the registration before it expires. If a presence based capability check is used (based on the DEFAULT DISCOVERY MECHANISM parameter specified in Table 82 in section A.1.10 see section 2.6), the device shall also publish its capabilities once it has registered.

2.4.6 P-CSCF discovery

Prior to any initial IMS registration the client shall discover the IP address of the P-CSCF as defined in [3GPP TS 23.228].

The P-CSCF discovery procedure shall be applied in accordance with the mode of operation of the RCS device:

- Devices enabled for VoLTE or VoWiFi shall select the P-CSCF as defined in [PRD-NG.102].
- Other devices shall select the P-CSCF from the LBO_P-CSCF_Address node of the IMS management object. If the P-CSCF AddressType of the P-CSCF address in the IMS management object indicates "FQDN" the device shall resolve the Fully Qualified Domain Name (FQDN) as defined in [RFC3263].
  For the protocol selection in [RFC3263], the device shall

- Take the SIP transport protocol settings in the IMS device management object (as defined in section 2.2.1 of [PRD-RCC.15]) into account if SIP Digest is to be used for authentication, i.e. SIPoUDP, SIPoTLS or SIPoTCP depending on the access network type (PS or Wi-Fi).
- Ignore the SIP transport protocol settings in the IMS device management object (as defined in section 2.2.1 of [PRD-RCC.15]) if AKA is to be used for Authentication. The device selects either UDP or TCP as defined in [3GPP TS 24.229].

If the P-CSCF discovery results in a list of P-CSCF addresses then the device shall select a new P-CSCF address for any initial registration in accordance with the priority indications (e.g. weight and priorities in DNS SRV) to support load distribution in the network.

2.4.7 IMS Flow Set Management

IMS flow set is defined in [3GPP TS 24.229]. It refers to the "flow" defined by the combination of transport protocol, client IP address and port and P-CSCF IP address and port used by the client and the network to exchange all SIP signalling related to a single IMS registration. This section details the requirements for an RCS client to manage the IMS flow set (i.e. a single Registration) in the network.

2.4.7.1 REGISTER Request Handling

The RCS client shall make use of the registration procedures as defined in [3GPP TS 24.229].

NOTE: [3GPP TS 24.229] specifies that the REGISTER shall be sent to the IP address and port obtained via the discovery procedure. If the device was unable to obtain a specific port, then the default port as specified in [RFC3261] will be used.
The client will send subsequent REGISTER and non-REGISTER requests to the IP address that is used for the initial REGISTER, unless the security mechanism requires the use of negotiated ports for the exchange of protected messages.

2.4.7.2 Non REGISTER Request Handling

The RCS client shall make use of the procedures for methods excluding the REGISTER method as defined in [3GPP TS 24.229].

The following addition applies to [3GPP TS 24.229] section 5.1.2A.1.1:

- The proper preload route header for methods excluding the REGISTER method shall be built only with the IP address learnt through the P-CSCF discovery procedure, i.e. a FQDN must not be used.

2.4.7.3 IMS Flow Set Termination

The RCS client should ensure that an IMS flow set is released in the network before the conditions for the existence cease to exist, e.g. prior to the release of the bearer the IMS flow set makes use of.

The IMS flow set shall be terminated by the client by sending a de-registration request to the network using the IMS flow set to be terminated. If there is one or more ongoing session on the IMS flow set, these shall be released first.

2.4.7.4 Loss of Connection to P-CSCF

If the connection to the P-CSCF fails (e.g. TCP time-out) the RCS client should select another P-CSCF address from the list of addresses obtained during the P-CSCF discovery in accordance with their priority indication.

If the P-CSCF discovery is based on the IMS management object and it contains one or more FQDNs, then the client shall invoke the [RFC3263] FQDN resolution anew. A different P-CSCF address shall be selected from the name resolution result in accordance with their priorities and weights.

The client shall then send a new initial registration using the new discovered P-CSCF address.

2.4.7.5 Loss of Connectivity

If a RCS client discovers that connectivity has been lost then it should attempt to re-establish the connection.

For a client that is not enabled for VoLTE or VoWiFi or a client enabled for VoLTE or VoWiFi that is not configured to share the registration between RCS and Multimedia Telephony, when connectivity has been resumed then;

- If the IP address has been changed and the transport protocol setting for the new connection (as derived from the Management Object defined in section 2.2.1 of [PRD-RCC.15] for the new access network type) is the same as for the lost connection and the IMS registration is not yet expired, the client shall perform a new initial registration to the P-CSCF address of the last IMS flow set in use.
- If the IP address has not been changed and the IMS registration is not yet expired, the client shall perform a re-registration using the existing IMS flow set if the IP address has not been changed and the IMS registration is not yet expired. To minimize the network impact in cases of unstable connectivity conditions the client should hold a minimum re-registration time in which no such re-registration requests are sent. The minimum re-registration time should be typically in the range of 3-5 minutes.
- In all other cases, the client shall perform a new P-CSCF discovery and a new initial registration.

NOTE: The registration or re-registration may trigger delivery of messages stored in the network during the absence of connectivity.

2.4.7.6 Detection of Connection Loss in RCS Clients with no Bearer Control Capabilities

RCS client implementations may have no capability to identify the cause of a connection loss due to missing bearer control capabilities.

These clients should identify the cause of a loss of connectivity via the following procedure.

- If the client detects a connection loss during a P-CSCF signalling interaction (e.g. TCP time-out), then it shall attempt the procedure defined in section 2.4.7.4.
- Only if a new IMS flow set is established with an alternative P-CSCF the client shall release the IMS flow set used for the old P-CSCF locally.
- If the connection establishment to the alternative P-CSCF or other targets in the network fails (e.g. DNS Server) then the client shall assume loss of connectivity and act as defined in section 2.4.7.5.
- If the client detects a connection loss during network interactions other than signalling with the P-CSCF (e.g. media connection, auto-configuration server) then the client shall assume loss of connectivity.

2.4.8 Loss of Registration

When the client receives a SIP response to a non-REGISTER request that is either:

- 403 Forbidden without a warning header, or
- 504 Server Timeout containing a P-Asserted-Identity URI matching a URI received during registration in Service-Route or Path header field and containing a 3GPP IM CN subsystem XML body with the <alternative-service> child element with the <type> child element set to “restoration” and the <action> child element set to “initial-registration”

(indicating loss of registration due to change of IP, expiration, network problem), the client shall attempt to register again using the procedure in section 2.4.5. When successful the client shall resend the request that caused the error response. If this fails for 5 consecutive retries though, no further attempt shall be made and an error should be shown to the user. For all services except One-to-One Chat, the retry procedures will also be stopped if it takes longer than 5 seconds. Also in that case an error message should be shown to the user.

NOTE: On receiving a 403 Forbidden response a client may before re-Registration first attempt to send a SIP request to his own URI and only re-Register if that request results in a 403 Forbidden response.

2.5 Addressing and identities

2.5.1 Overview

Telephone numbers in the legacy address book must be usable (regardless of whether RCS contacts have been enriched or not) for the identification of contacts of incoming and outgoing SIP requests.

Also, RCS users, especially in Enterprise segments, may be assigned a non MSISDN based identity. The RCS client would in that case be provisioned with only the appropriate SIP URI parameter as seen in section 2.2.1.1.2 of [PRD-RCC.15], leaving the tel URI parameter empty.

Consequently, an RCS enabled terminal’s address book should also be able to store alphanumeric SIP URIs as part of a contact’s details.

NOTE1: The handling of identities described in this section applies also to IP Voice Calls [PRD-IR.92] and [PRD-IR.51]. The functionality described here comes in addition to the functionality described in the related Permanent Reference

Documents (PRDs), but not in conflict with them, e.g. the alias handling described in section 2.5.3.4.

NOTE2: The identification in Common Profile for Instant Messaging (CPIM) headers is discussed in section 3.3 and 3.4.

2.5.2 Device Incoming SIP Request

2.5.2.1 From/P-Asserted-Identity

For device incoming SIP requests, the address(es) of the contact are, depending on the type of request, provided as a URI in the body of a request or contained in the *P-Asserted-Identity* and/or the *From* headers. If the *P-Asserted-Identity* header is present, the *From* header will be ignored. The only exception to this rule is when a request for Chat or Standalone Messaging includes a *Referred-By* header (it is initiated by Messaging Server for example in a store and forward use case as described in 3.3.4.1.4), thereby the *Referred-By* header should be used to retrieve the originating user instead.

The receiving client will try to extract the contact’s phone number out of the following types of URIs:

- tel URIs (telephone URIs, for example tel:+1234578901, or tel:2345678901;phone-context=<phonecontextvalue>)
- SIP URIs with a “user=phone” parameter, the contact’s phone number will be provided in the user part (for example sip:+1234578901@operator.com;user=phone or sip:1234578901;phone-context=<phonecontextvalue>@operator.com;user=phone)

Once the MSISDN is extracted, it will be matched against the phone number of the contacts stored in the address book. If the received URI is a SIP URI but does not contain the “*user=phone*” parameter, the incoming identity should be checked against the SIP and tel URI address of the contacts in the address book instead.

If more than one *P-Asserted-Identity* is received in the message, all identities shall be processed until a matched contact is found.

2.5.2.2 In-call SIP requests

2.5.2.2.1 Caller: relating the ongoing call with in-call incoming SIP requests

The destination identity of the telephony call that the caller dials or gets from his address book and the originator identity of any in-call incoming request may be in various formats. The client of the caller shall, therefore, apply the following matching mechanism to determine whether an incoming request relates to the ongoing call:

1. If both the destination identity of the telephony call and the originator identity of the in-call incoming request are phone numbers in international format, the client of the caller shall compare all digits of the provided numbers to determine whether they match.
2. If any of the identities is not in international format, the client of the caller shall apply an enhanced matching mechanism between the destination identity from the telephony call and the originator identity of the incoming request, e.g. by comparing

the 7 digits starting from the end of the number. It is left to the client implementation to apply an even more enhanced matching algorithm to decrease the probability of false matches.

The client shall consider the identities to be in international format if

- For a CS or multimedia telephony outgoing call, the digits dialled or taken from the address book start with a “+”.
- Tor an incoming request, the P-Asserted-Identity of the SIP request contains either:
  - a tel URI starting with a “+” without phone-context i.e. a global number, or
  - a SIP URI with user part starting with a “+”, a user=phone parameter and without a phone-context parameter in the user part.

Examples:

The destination identity of the outgoing telephony call: **+447123456789** (display string for an international format number).
The originator identity of the incoming request: **+447123456789**
➔ Matching result: Successful

When the applied enhanced matching algorithm is based on the 7 digits starting from the end of the number:
The destination identity of the outgoing telephony call: 0712**3456789** (non-international format).
The originator identity of the incoming request: +44712**3456789**
➔ Matching result: Successful

2.5.2.2.2 Callee: relating the ongoing call with in-call incoming SIP requests

The originator identity of the telephony call and the originator identity of any in-call incoming request may be provided in various formats both in the home networks and when roaming. The client of the callee shall therefore apply the following matching mechanism to determine whether an incoming request relates to the ongoing call:

1. If both the originator identity of the telephony call and the originator identity of the in-call incoming request are phone numbers in international format, the client of the callee shall compare all digits of the provided numbers to determine whether they match.
2. If any of the originator identities is not in international format, the client of the callee shall apply an enhanced matching mechanism between the originator identity from the telephony call and the originator identity of the incoming request, e.g. by comparing the 7 digits starting from the end of the number. It is left to the client implementation to apply an even more enhanced matching algorithm to decrease the probability of false matches.

The client shall consider the identities to be in international format if

- For a CS incoming call, the Type Of Number (TON) of the Calling Party BCD Number is set to “international” as defined in [3GPP TS 24.008].

- For a multimedia telephony incoming call, the P-Asserted-Identity of the SIP INVITE request contains either:
  - tel URI starting with a “+” without phone-context i.e. a global number or
  - SIP URI with user part starting with a “+”, a user=phone parameter and without a phone-context parameter in the user part.
- For an incoming request, the P-Asserted-Identity of the SIP request contains either:
  - a tel URI starting with a “+” without phone-context i.e. a global number or
  - a SIP URI with user part starting with a “+”, a user=phone parameter and without a phone-context parameter in the user part.

Examples:

The originator identity of the incoming telephony call: **+447123456789** (display string for an international format number).
The originator identity of the incoming request: **+447123456789**
➔ Matching result: Successful

When the applied enhanced matching algorithm is based on the 7 digits starting from the end of the number:
The originator identity of the incoming telephony call: 00644712**3456789** (non-international format).
The originator identity of the incoming request: +44712**3456789**
➔ Matching result: Successful

2.5.3 Device Outgoing SIP Request

2.5.3.1 Identification of the target contact

If the target contact contains a SIP or tel URI the value shall be used by the RCS client when generating the outgoing request even if an MSISDN is also present for the contact. This applies to the SIP Request-URI and the “*To*” header (as defined in [3GPP TS 24.229]) for 1-to-1 communication, including the URIs used in the recipient list included in outgoing SIP requests for Group Chat.

If no SIP or tel URI is present the RCS client shall use the telephone number (in local format for example *0234578901* or international format *+1234578901*) set in the address book or a dial string entered by the user.

If the target number is an international-format telephone number, the device shall be able to send it as tel URI (for example “*tel:+12345678901*”) as defined in [RFC3966].

If the target number is a non-international format telephone number, the RCS client shall be able to send it as tel URI with a phone-context value set as defined in [3GPP TS 24.229] for home local numbers (for example *tel:0234578901;phone-context=<home-domain-name>*).

2.5.3.2 In-call SIP requests

2.5.3.2.1 Caller: addressing SIP requests towards the callee

The destination identity of the telephony call that the caller dials or gets from the address book may be in various formats. The client of the caller shall, therefore, apply the following principles for addressing the callee when triggers in-call SIP requests:

1. If the destination identity of the telephony call is in international format, the client of the caller shall use this information for addressing in-call SIP requests towards the callee.
2. If the destination identity of the telephony call is not in international format, the client of the caller shall use geo-local numbering of the destination identity of the telephony call for addressing in-call SIP requests towards the callee. If the request fails, the client of the caller shall attempt to correlate the destination identity of the telephony call with his local identity records acquired from incoming SIP requests received in a window prior to the call and/or during the call using an enhanced matching mechanism between the destination identity from the telephony call and the incoming SIP requests, e.g. by comparing the 7 digits starting from the end of the number. It is left to the client implementation to set the time length of the window and apply an even more enhanced matching algorithm to decrease the probability of false matches.

    a) If there is successful matching, the client of the caller shall use the “matched” destination identity from his local identity records for addressing in-call SIP requests towards the callee.
    b) If there is no successful matching, the client of the caller shall use the destination identity from the telephony call that the caller dials or gets from his address book for addressing in-call SIP requests towards the callee. The client of the caller shall continue applying the enhanced matching mechanism for any in-call incoming SIP request until it matches the destination identity from the telephony call with the originator identity from an in-call incoming SIP request. Once there is a successful matching it shall from then on use the “matched” originator identity from the SIP request for addressing any future in-call SIP requests towards the callee.

The client shall consider the identities to be in international format if

- for a CS or multimedia telephony outgoing call, the digits dialled or taken from the address book start with a “+”.

Examples:

> The destination identity of the outgoing telephony call: +447123456789 (display string for an international format number).
> The client of the caller uses the destination identity from the telephony call for addressing in-call SIP requests towards the callee.
> The enhanced matching mechanism does not apply.
>
> The destination identity of the outgoing telephony call: 0712**3456789** (non-international format).

The client of the caller shall use geo-local numbering of the destination identity of the telephony call for addressing in-call SIP requests towards the callee: tel:07123456789;phone-context=geolocal.<homedomain>, where <homedomain> needs to be replaced with the home network domain name as configured by the device (as per section 2.2.3 of [PRD-IR.92]).
If the in-call SIP request fails, the client shall apply the enhanced matching mechanism.
The originator identity of incoming SIP request: +447123456789.
➔ Matching result: Successful

2.5.3.2.2 Callee: addressing SIP requests towards the caller

The originator identity of the telephony call may be provided in various formats both in the home networks and when roaming. The client of the callee shall therefore apply the following principles for addressing the caller when triggers in-call SIP requests:

1. If the originator identity of the telephony call is in international format, the client of the callee shall use this information for addressing in-call SIP requests towards the caller.
2. If the originator identity of the telephony call is not in international format, the client of the callee shall use geo-local numbering of the originator identity of the telephony call for addressing in-call SIP requests towards the caller. If the request fails, the client of the callee shall attempt to correlate the originator identity of the telephony call with his local identity records acquired from incoming SIP requests received in a window prior to the call and/or during the call using an enhanced matching mechanism between the originator identity from the telephony call and the incoming SIP requests, e.g. by comparing the 7 digits starting from the end of the number. It is left to the client implementation to set the time length of the window and apply an even more enhanced matching algorithm to decrease the probability of false matches.

    a) If there is successful matching, the client of the callee shall use the “matched” originator identity from his local identity records for addressing in-call SIP requests towards the caller.
    b) If there is no successful matching, the client of the callee shall use the originator identity from the telephony call for addressing in-call SIP requests towards the caller. The client of the callee shall continue applying the enhanced matching mechanism for any in-call incoming SIP request until it matches the originator identity from the telephony call with the originator identity from an in-call incoming SIP request. Once there is a successful matching it shall from then on use the “matched” originator identity from the SIP request for addressing any future in-call SIP requests towards the caller.

The client shall consider the identities to be in international format if

- for a CS incoming call, the Type Of Number (TON) of the Calling Party BCD Number is set to “international” as defined in [3GPP TS 24.008].
- for a multimedia telephony incoming call, the P-Asserted-Identity of the SIP INVITE request contains either:

- a tel URI starting with a "+" without phone-context i.e. a global number or
- a SIP URI with user part starting with a "+", a user=phone parameter and without a phone-context parameter in the user part.

Examples:

The originator identity of the incoming telephony call: +447123456789 (display string for an international format number).
The client of the callee uses the originator identity from the telephony call for addressing in-call SIP requests towards the caller.
The enhanced matching mechanism does not apply.

When the applied enhanced matching algorithm is based on the 7 digits starting from the end of the number:
The originator identity of the incoming telephony call: 006447123**3456789** (non-international format).
The client of the callee shall use geo-local numbering of the originator identity of the telephony call for addressing in-call SIP requests towards the caller: tel: 006447123456789;phone-context=geolocal.<homedomain>, where <homedomain> needs to be replaced with the home network domain name as configured by the device (as per section 2.2.3 of [PRD-IR.92]).
If the in-call SIP request fails, the client shall apply the enhanced matching mechanism.
The originator identity of incoming SIP request: +44712**3456789.**
➔ Matching result: Successful

2.5.3.3 Self-Identification to the network and the addressed contact

When generating an outgoing non-REGISTER request, the RCS client shall populate the *From* header field and may populate the *P-Preferred-Identity* header field with a SIP or tel URI which has been received in the *P-Associated-URI* header field returned in the 200 OK to the SIP REGISTER. If both a SIP URI and a tel URI are available to the RCS client, the tel URI should be used.

2.5.3.4 User alias

The user shall be able to specify an alias or a username for RCS services. This information will be sent when establishing a communication service with another user so they are able to receive additional information (i.e. beyond than just a MSISDN), if the originating user is not in the receiver's address book. This scenario will likely be common with Group Chat sessions.

This alias information will be set in the *From* header of the SIP request as the display name and in a Group Chat also in the CPIM *From* header as the formal name.

When receiving a request, the RCS client device shall follow the rules explained in section 2.5.2.1 and extract the MSISDN or SIP URI. To avoid spam and identity manipulation, the receiver shall check the identity of the calling user against the address book. If the user is not in the address book, the alias information must then be used to provide more information about the calling user while clearly displaying in the User Interface (UI) that the identity is unchecked and it could be false. Otherwise the name of the contact in the address book shall be used instead.

2.6 Capability and new user discovery mechanisms

2.6.1 Capability discovery

The capability or service discovery mechanism is a process which enhances service usability by allowing a user to understand the subset of RCS services available to access and/or communicate with their contacts, at certain points in time.

When available, the RCS specification provides two alternative mechanisms to perform the capability discovery:

- SIP OPTIONS exchange (section 2.6.1.1):
  - The SIP OPTIONS end-to-end message is used both to query the capabilities (services which the other user has available) of the target contact and to pass the information about which capabilities are supported by the requester. Using this method, both users get updated information in a single transaction. However, users are allowed to hide their service capabilities to some undesired contacts. To this end, clients may support a locally stored capability blacklist which could be configured manually by the user or be generated automatically by setting any contacts that do not exist in the user’s address book as the blacklisted contacts. This blacklist is different with the blacklist that is used for chat, file transfer and content share services. If a SIP OPTIONS request is received from a capability blacklisted contact, the client should answer the request with an empty 200 OK which does not include any of the service tags used by RCS services (see Table 21).
  - This method requires a specific application server (Options-AS) in the network to provide multidevice support and, potentially, include optimisations.
- Presence (section 2.6.1.2):
  - In this case, instead of performing an end-to-end transaction, the capabilities are queried against a server using the standard OMA SIMPLE Presence procedures which are described in detail in section 2.6.1.2.
  - Consistent with the previous paragraph and the OMA SIMPLE Presence procedures, this method requires both a Presence and a XDM server in the network.

The default discovery mechanism is configured in the device using the configuration parameter CAPABILITY DISCOVERY MECHANISM (see Annex A section A.1.10).

In accordance with the principle of interoperability between RCS networks and devices, two mechanisms are provided to secure the interoperability between the mechanisms presented before:

- Co-existence based in a common device stack (section 2.6.1.3):
  - The interoperability is provided via a device implementation and, consequently, no additional network elements are required.

  - The principle of interoperability is that all devices support SIP OPTIONS exchange either as a default or a device fall-back mechanism (when the presence query fails for a particular user).

- Coexistence based in network interworking (section 2.6.1.3.3):

  - Network Interworking is required between Service Providers that do not support SIP OPTIONS exchange (as the default method or as a device fall-back mechanism) and those Service Providers that use SIP OPTIONS as the default discovery mechanism.
  - Interoperability is achieved by deploying a network based interworking function which translates requests and responses between the SIP OPTIONS and presence-based capability discovery mechanisms.

To guarantee that Service Providers choosing the network interworking approach do not experience situations whereby the device fall-back mechanism to SIP OPTIONS occurs, a new parameter (CAPABILITY DISCOVERY VIA COMMON STACK) is defined. The device fall-back mechanism only occurs if this parameter is set to 1 (see Annex A sections A.1.10 and A.2.8 for further reference).

2.6.1.1 Capability discovery process through SIP OPTIONS message

One of the available mechanisms for capability discovery is based on the exchange of a SIP OPTIONS request, a peer-to-peer message exchanged between clients.

When a SIP OPTIONS message is sent from User A to User B, User A shall handle the response as described in the following table:

Response	User B was a known RCS user before	User B was not a known RCS user before
200 OK including at least, one of the tags assigned to the RCS Services (see Table 21) Returned when User B is an RCS user and is currently registered	User B remains an RCS user The capabilities returned in the 200 OK response (using tags as described in Table 21) are considered as the current communication options with User B	User B is marked as an RCS user The capabilities returned in the 200 OK response (using tags as described in section Table 21) are considered as the current communication options with User B

Response	User B was a known RCS user before	User B was not a known RCS user before
200 OK not including any of the tags used by RCS services (see Table 21) Returned when User B is registered, but not with an RCS client or User A is in the capability blacklist of User B	User B is not considered as an RCS user any longer Only the non-RCS communication services (e.g. voice calls, SMS, MMS, etc.) are indicated as available[4]	No change in User B's status Only the non-RCS communication services (e.g. voice calls, SMS, MMS, etc.) are indicated as available
480 TEMPORARY UNAVAILABLE or 408 REQUEST TIMEOUT Returned by the network if User B is an IMS (and potentially thus an RCS) user, but is currently not registered	User B remains an RCS user but only the capabilities available to an offline contact are offered	No change in User B's status Only the non-RCS communication services (e.g. voice calls, SMS, MMS, etc.) are indicated as available
404 Not Found or 604 Does Not Exist Anywhere	User B is not considered as an RCS user any longer Only the non-RCS communication services (e.g. voice calls, SMS, MMS, etc.) are indicated as available	No change in User B's status Only the non-RCS communication services (e.g. voice calls, SMS, MMS, etc.) are indicated as available
Any other Final response returned by the network	User B remains an RCS user with unchanged capabilities. NOTE: The client treats the final response as described in [3GPP TS 24.229].	No change in User B's status

Table 5: Options response handling

In some cases sending an OPTIONS request is not required as the last SIP OPTIONS exchange took place just before the communication was set up (e.g. to send a SMS message, the user went to the address book, selected a user [SIP OPTIONS exchange takes place] and chooses to send a SMS message).

[4] Note that this means that an AS like the OPTIONS-AS described in section 2.6.1.1.5 would have to include the IM capability in the response if the user has multiple devices sharing the same IMS identity some of which are not RCS capable. When including this tag though in situations where none of the RCS capable devices is online, it shall also include the *automata* tag defined in [RFC3840] to indicate that this response does not originate from an end user device.

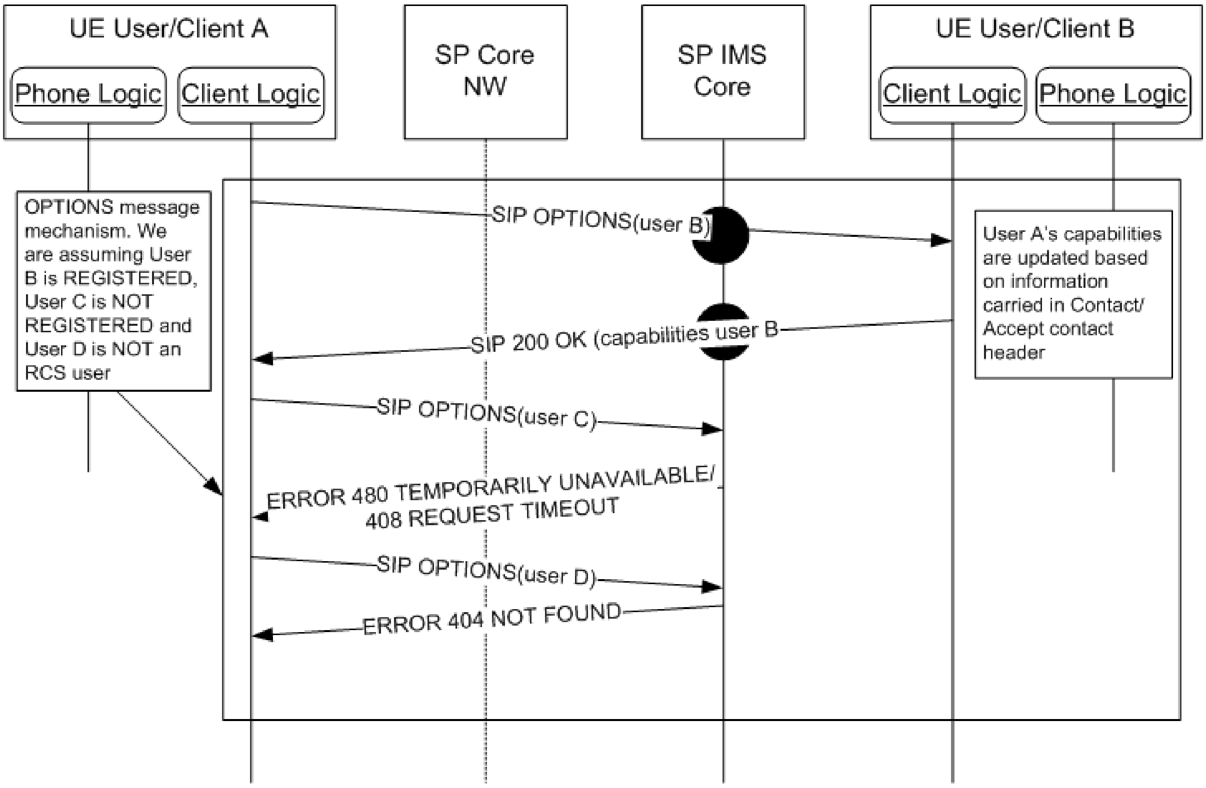

Figure 6: Capabilities discovery via SIP OPTIONS message

2.6.1.1.1 SIP OPTIONS message extension to support capability discovery

The RCS (Release 1 to 4) specifications only provide a mechanism to exchange the capability status (based on a SIP OPTIONS exchange) related to the Image and Video Share services during a call (associated with the capability query procedure described in [PRD-IR.74] and [PRD-IR.79]). This mechanism is based on the use of tags transported in the *Contact* header field for the SIP OPTIONS and its responses:

- The tags corresponding to the set of functionalities supported by the requesting terminal at the time this request is made are carried in the Contact header field of the SIP OPTIONS message.
- The tags corresponding to the subset of the functionalities that are supported by the receiver are included in the Contact header of the 200 OK responses.

When the SIP OPTIONS is sent as part of an ongoing voice call, as per [PRD-IR.74] and [PRD-IR.79], the Accept-Contact header shall be handled as described in [PRD-IR.74] and [PRD-IR.79].

As described in [PRD-IR.74] and [PRD-IR.79], to have a Session Description Protocol (SDP) body in an OPTIONS request message is optional. It is not encouraged behaviour to insert it into this message. In RCS, the SIP OPTIONS request shall NOT contain an SDP body.

The mechanism described above is extended to be used not only for the exchange of capabilities for real-time services but also to query in real time to exchange the capabilities/services supported by both the requester and the receiver.

When the Service Provider disables the capability discovery mechanism the network shall return a SIP 200 OK including any of the agreed interworking service tags which are supported by User B as described in Table 21. If User B is currently not registered or is not a RCS user, the network shall respond in accordance with section 2.6.2.1 Discovery via OPTIONS message.

2.6.1.1.2 Extensions to the existing tags

Consequently, with the RCS Release 1-4 specifications, the following tags can be employed to identify Image and Video Share service capabilities during a call:

RCS service	Tag
Image Share	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.gsma-is"
Video Share	+g.3gpp.cs-voice

Table 6: Standard RCS Release 1-4 SIP OPTIONS tags

When used in SIP OPTIONS exchanges these Image and Video Share capabilities can only be sent during an active call and are included only if the exchange takes place between the users in the active call. However, Broadband Access devices should include these capabilities in an OPTIONS response even if they are not in an active call.

To support the full service discovery functionality presented in this document, it is necessary to extend the tag mechanism by adding the following service tags:

- As interoperability between the different technical implementations for Chat and File Transfer services is assumed, the following tags are employed for the Chat and File Transfer services:

RCS service	Tag
Chat	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcse.im"
File Transfer via MSRP	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcse.ft,urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.ftthumb"[5]
File Transfer via MSRP Store and Forward	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.ftstandfw"
File Transfer via HTTP	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.fthttp"

Table 7: SIP OPTIONS tags for Chat and File Transfer

The File Transfer via MSRP Store and Forward capability shall only be provided when File Transfer via MSRP is enabled through the PROVIDE FT configuration parameter described in section A.1.5 and the Chat Technology is set to OMA CPM

[5] The urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.ftthumb IARI is included to provide backwards compatibility towards older versions of RCS

using the CHAT MESSAGING TECHNOLOGY configuration parameter defined in section A.1.4.3.

- Add a tag for IP based standalone text and multimedia messaging:

RCS service	Tag
IP Based Standalone messaging	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.msg,urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.largemsg"

Table 8: SIP OPTIONS tag for standalone messaging

- Add a tag for Social Presence Information:

RCS service	Tag
Social presence information	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcse.sp"

Table 9: SIP OPTIONS tag for Social Presence Information

- Add tags for IP Voice and Video Call services:

RCS service	Tag
IP Voice Call (as per MMTEL)	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.mmtel"
IP Video Call(as per MMTEL)	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.mmtel”;video
RCS IP Voice Call	+g.gsma.rcs.ipcall
RCS IP Video Call	+g.gsma.rcs.ipcall;video
RCS IP Video Call where video media cannot be removed by the user	+g.gsma.rcs.ipvideocallonly

Table 10: SIP OPTIONS tags for IP Voice and Video Call

NOTE1: When a device supports both IP Voice Call and IP Video Call, the feature tags *+g.3gpp.icsi-ref="urn:urn-7:3gpp-service.ims.icsi.mmtel"* and *+g.gsma.rcs.ipcall* are only included once in the OPTIONS request/response.

A device shall provide in the SIP OPTIONS requests and responses only one of the *RCS IP Voice Call*, *RCS IP Video Call* and *RCS IP Video Call where video media cannot be removed* capabilities depending on whether according to the PROVIDE RCS IP VOICE CALL and PROVIDE RCS IP VIDEO CALL configuration parameters defined in section A.1.12 in the currently used radio technology respectively only RCS IP Voice Calls, both RCS IP Voice Calls and RCS IP Video Calls or only RCS IP Video Calls are allowed. Due the combining of the capabilities of different devices (see section 2.6.1.1.5), it may happen though that multiple of those capabilities are received. If the RCS IP Video Call where video media cannot be removed capability is received in combination with the RCS IP Video Call and/or the RCS IP Voice Call capability, towards that contact both RCS IP Voice Call and RCS IP Video Calls may be successfully initiated as described in sections 3.8 and 3.9 when those services are allowed according to the local configuration and used radio technology.

NOTE2: RCS IP Calls and IP Calls per MMTEL can technically interoperate. Therefore, an RCS IP Call client is allowed to answer an incoming IP Call (and vice versa), and shall process the received capability information accordingly.

If the device is configured to support only the RCS IP Video Call services as described in section A.1.12 (i.e. RCS IP Voice Call initiation is not allowed and a downgrade by the user of an RCS IP Video Call to an RCS IP Voice Call is not allowed), it shall include the *+g.gsma.rcs.ipvideocallonly* tag described in section 2.4.4 in the OPTIONS exchanges when in coverage where RCS IP Video Calls are supported.

The applicable RCS IP Call feature tags are only included in the OPTIONS exchanges when the device is in coverage where it is configured to support RCS IP Voice and Video Calls.

NOTE3: In case of interworking between two Service Providers, the validity of the IP Video Call capability tag highly depends on the end-to-end interconnection chain.

- Add tags for the Geolocation services:

RCS service	Tag
Geolocation PUSH	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.geopush"
Geolocation PULL	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.geopullft"

Table 11: SIP OPTIONS tags for geolocation services

- To support the service functionality presented in [PRD-RCC.20] it is necessary to extend the tag mechanism by adding the following service tags:
  - Add a tag for Call composer service:

RCS service	Tag
Call composer	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.gsma.callcomposer"

Table 12: SIP OPTIONS tags for call composer

- Add a tag for Post-call service:

RCS service	Tag
Post-call	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.gsma.callunanswered"

Table 13: SIP OPTIONS tags for Post-call

- Add a tag for Shared Map service:

RCS service	Tag
Shared Map	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.gsma.sharedmap"

Table 14: SIP OPTIONS tags for Shared Map

When used in SIP OPTIONS exchanges this Shared Map capability can only be sent during an active call and is included only if the exchange takes place between the users in the active call.

- Add a tag for Shared Sketch service:

RCS service	Tag
Shared Sketch	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.gsma.sharedsketch"

Table 15: SIP OPTIONS tags for Shared Sketch

When used in SIP OPTIONS exchanges this Shared Sketch capability can only be sent during an active call and is included only if the exchange takes place between the users in the active call.

Unless specified in other sections (e.g. section 2.4.4), the new tags defined in this section are defined for use in SIP OPTIONS exchanges only and the standard tags defined in the supporting PRDs and endorsement documents shall be used to identify the services in the rest of relevant SIP transactions (e.g. *+g.oma.sip-im* for Chat implementation based on OMA SIMPLE IM as per [RCS-SIMPLEIM-ENDORS]). It should also be noted that in some cases, the tags employed in the SIP OPTIONS exchange match the standard tags.

A device should also add to the Contact header field the same feature tags used at SIP Registration if not already included in the OPTIONS request/response for capability exchange and if they are part of the capabilities supported by the device at this time. This mainly applies to the +g.oma.sip-im feature tag which is used at SIP Registration and in SIP transactions but not used to identify a service capability.

Finally, it should be taken into account that when several IMS Application Reference Identifier (IARI) tags or several ICSI tags are included in an OPTIONS request, consistently with [RFC3840], IARI tags or ICSI tags shall be concatenated using commas as described in the example below:

+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcse.im,urn%3Aurn-7%3A3gpp-application.ims.iari.rcse.ft"

Table 16: IARI tag concatenation format example

2.6.1.1.3 Future extensions to the mechanism

In addition to the aforementioned additions and to allow:

- A Service Provider (or group of Service Providers) to deploy additional services which can benefit from the RCS discovery mechanism, an additional tag format is defined:
  - +g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.mnc<mnc>.mcc<mcc>.<service name>"
  - Valid examples are:
    - +g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.mnc001.mcc214.serviceA"
    - +g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari rcs.mnc680.mcc310.serviceB"

The service name is decided by the each Service Provider. The only requirement for a Service Provider following this approach is to include these tags in the relevant interoperability agreements with other Service Providers.

- A third-party to deploy an Extension (e.g. through device API as per [PRD-RCC.53]) which can benefit from the RCS discovery mechanism, an additional tag format is defined:
  - +g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.ext.<identifier>"
    - Where <identifier> is an identifier (encoded in base64 as per [RFC4648]) uniquely identifying the application. The way the system ensures uniqueness of the value of the identifier is out of scope of this specification. The length of this field shall not exceed 64 bytes.
  - A valid example is:
    - +g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.ext.A5TgS99bJloIUIh1209SJ82B21m87S1B87SBqfS871BS8787SBXBA3P45wjp63tk”

NOTE: A Service Provider may deploy Extensions as a third-party.

The use of those Extensions for actual third-party usages requires a full functioning framework, including a secure management of the Extensions on the devices. If this framework is not in place, Clients shall not enable third parties to use this tag.

These IARIs are only exchanged in capability discovery and updates when the Extensions make use of the RCS infrastructure to communicate (as per section 3.12.4.2).

RCS service	Tag
Service Provider specific service	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs .mnc<mnc>.mcc<mcc>.<service name>"
Third-Party specific service	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs .ext.<identifier>"

Table 17: SIP OPTIONS tag proposal for future lines of work

2.6.1.1.4 UI integration optimisations

In addition to the optimisations to minimise the traffic generated by the SIP OPTIONS exchanges when possible, there are two additional optimisations related to the discovery mechanism integration on the UI that should be taken into account:

- The round trip time for a SIP OPTIONS exchange (send and receive response) is expected to range between under 1-2 seconds. Taking this into account, the UI has to be optimised to minimise the impact of this exchange delay.
- When sending the SIP OPTIONS messages to several users (for example, during first time registration or when polling), it is recommended to employ a non-aggressive strategy and allow time between each exchange to:
    - Minimise potential network impact
    - Avoid any impact on the user experience (for example a slower UI, blockings and so on)

NOTE: In this case this specification does not specify the specific mechanisms which should be implemented leaving space to Original Equipment Manufacturers (OEMs) and third parties to drive innovative and differentiated solutions, which distinguish their products from competitors.

2.6.1.1.5 Multidevice support: Options-AS

Ultimately, the choice of supporting multiple devices for a single user is decided by each Service Provider. The considerations contained in this section will only apply to those Service Providers willing to include RCS multidevice support in their networks.

In a multidevice scenario, when the user is registered to the IMS CORE with various devices using the same URI (that is the same implicit registration set), the OPTIONS exchange will return incomplete information:

- The capabilities contained in the OPTIONS message refer only to the originating device (that is the originating user may be logged in with the same URI in several devices).
- The IMS Core, depending on the configuration, either sends the OPTIONS message to the device that first registered to the IMS CORE or forks the OPTIONS to all of the registered devices. In any case, only the first response is passed back to the requester, discarding the others. In other words, the capabilities returned in the OPTIONS response will be from only one of the user’s devices.

The preferred implementation for handling the OPTIONS in a multidevice environment is left to the Service Provider's discretion. The only requirement is that it should not impact the terminal side (that is there will be no changes on the client side). A possible solution for extending the OPTIONS mechanism to a multidevice scenario is to include a custom AS implementing the following logic:

- A trigger will be setup in the IMS CORE to send all of the OPTIONS from an RCS user to the AS.
- The AS will fork the OPTIONS request to all of the RCS user's registered devices and will aggregate all of the capabilities returned into one OPTIONS response if the forking is not already implemented by the IMS core network.
- Once the responses from the different devices are received, the AS will aggregate all the capabilities from the replies and send them back to the caller.
- Even if not all of the replies have been received in less than a configurable amount of time, the AS will return the aggregated information received so far.

NOTE: the recommendation is to set the value to optimise the UX on the terminal.

- Capabilities shall be aggregated to provide the response to an incoming SIP OPTIONS request. For outgoing requests, it is up to the Service Provider's policy to aggregate the capabilities.

NOTE: Similar procedures may at the service provider's discretion also be applied at originating side to aggregate the capabilities of all the user's devices in the OPTIONS request.

To implement this feature, an application server should be able to uniquely identify each user device to perform the forking of the OPTIONS message and to intercept and process the responses. The mechanism to have these individual identities (a GRUU or sip.instance feature tag) is covered in section 2.11.2.

While multidevice support is an option for each Service Provider to decide whether or not it is supported, the RCS capability discovery mechanism based on the SIP OPTIONS message is a mandatory requirement and the behaviour will be the one specified previously to ensure seamless interworking between Service Providers.

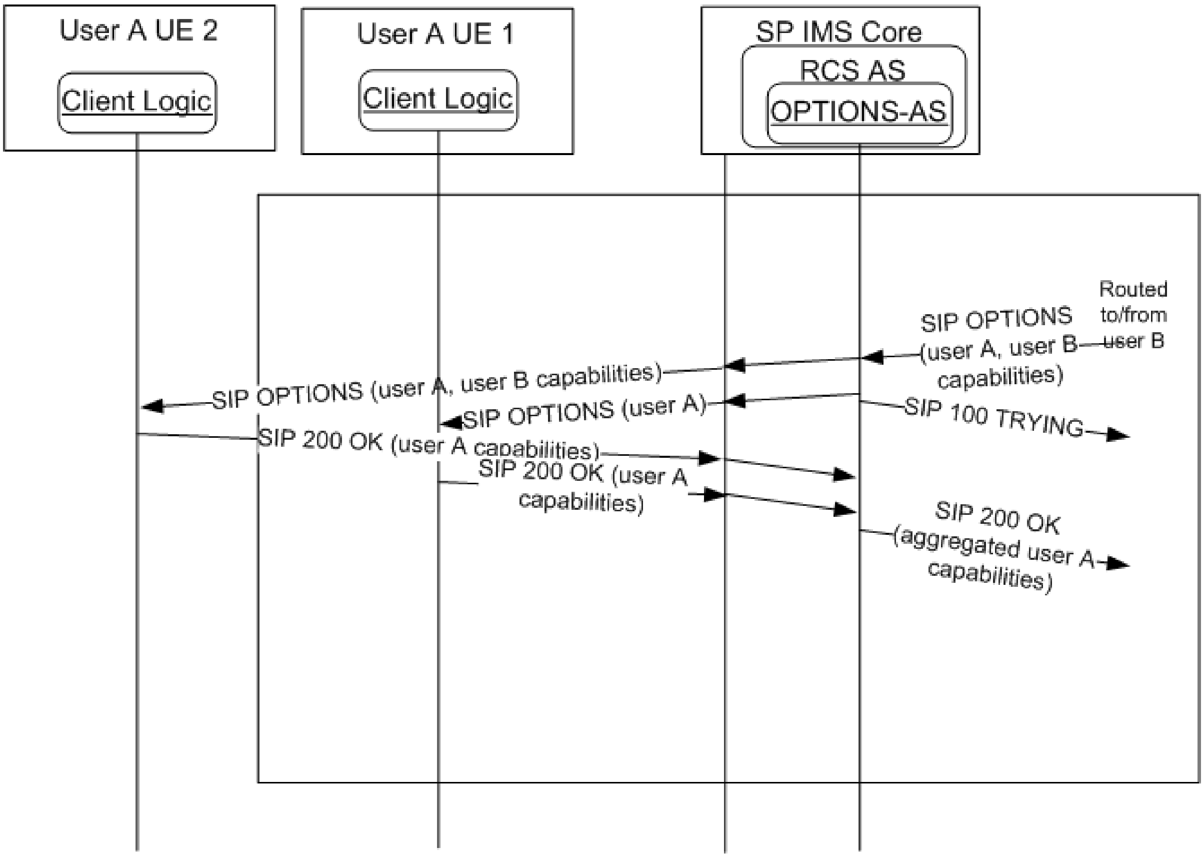

Figure 7: Options application server: Capability aggregation on SIP OPTIONS request

2.6.1.2 Capability discovery via presence

2.6.1.2.1 General Overview

As an alternative to the SIP OPTIONS-based mechanism presented in the previous section, a Service Provider deploying a Presence Server may provide the capability discovery mechanism via presence. The service capabilities are then realised using the "*Service*" part of the Presence Data Model. This part is described in section 2.6.1.2.5.

2.6.1.2.2 Publication of the Service Capabilities

The capabilities are announced in a Presence document that is published by using the SIP PUBLISH method as defined in [Presence]. When the terminal is started, the client then sends a SIP PUBLISH request containing the capabilities (see section 2.6.1.2.5). This SIP PUBLISH request shall not include an *Expires* header field.

The publication is maintained in the Presence Server whenever the application is running by sending a refresh request before it expires.

If changes are required in the published capabilities (for example, due to the behaviour specified in sections 2.6.2 and 2.6.3), a presence modify request is sent using the '*Sip-If-Match*' header according to [Presence]. When the client/device is switched off, it shall remove the published capabilities before unregistering according to the procedure defined in [RFC3903] (i.e. by sending a SIP PUBLISH request without a body including the '*Sip-If-Match*' header and an *Expires* header set to 0).

2.6.1.2.3 Service Capabilities Retrieval

Service capabilities of an RCS user can be retrieved by another RCS user via a presence subscription issued by their client, providing the pertaining Presence Authorisation rules allow him to do so. The templates provided in sections 2.14.1 and 3.7.4.5.2 allow this for the authorised users. An RCS user is, therefore, allowed to retrieve the service capabilities of contacts when they have an established Social Presence relationship.

RCS users may also retrieve the service capability information of contacts with which they have not established a Social Presence relationship by means of Anonymous Fetch operations issued by their client (as described in section 7.1 of [PRESENCE]). This will result in a single NOTIFY request indicating the service capabilities of that contact. This information shall then be cached in the client as described in section 2.6.4. The Anonymous Fetch operation shall be supported in clients.

If an RLS-URI (Resource List Server URI, see Annex A section A.1.2.1) has been provisioned, a client shall use an Anonymous Fetch request using a request-contained list if the client has to query the capabilities of multiple users at once (e.g. during a poll). In this case it shall do so according to section 5.2.1.2.2 of [Presence2.0_TS].

If only a single contact needs to be queried, an individual fetch shall be done instead even if an RLS-URI has been configured.

2.6.1.2.3.1 General Processing Rules to Ensure Backwards Compatibility

To maintain enough flexibility and not to impose potentially sub-optimal technical choices on future RCS versions, the parsing of the capabilities in an RCS client should be sufficiently robust. First the watcher should apply the processing rules defined in [Presence2.0_DDS] and if then there are still multiple elements the watcher shall follow the guidelines in the RCS presence parsing presented below:

- Unknown or unsupported elements and tuples could be present in the document. In that case they should be ignored.
- Unknown service identifiers (Service-Id) could be present in the document. Tuples containing those should be ignored.
- Unknown service versions of known services could be present in the presence document. Tuples containing those should be ignored.
- The same service could occur multiple times in the presence document with different contact addresses. To cope with this case, the following behaviour shall be used for displaying and using the tuples:

  - If one of the tuples contains a contact address that corresponds to the presentity about which the presence document was received, all others shall be ignored.
  - Tuples that contain a contact (address) element which corresponds to another presentity (that is another contact in the contact-list of the user or another tel URI) shall be ignored.
  - Tuples containing contact elements with types of addresses that are not supported by the client for that service shall be ignored (for example messaging using an e-mail address while e-mail is not supported by the client).

    - If after applying the above rules, there are still multiple non-ignored tuples remaining for the service, all but the first shall be ignored.
    - If after applying the above rules, there is a non-ignored tuple remaining, the service behaviour shall be as follows

        - The capability to use the service for communication with the contact shall be announced to the user
        - If the remaining tuple contained no contact address or it matched the one of the presentity, the presentity's address will be used for setting up communication using that service
        - Otherwise the address contained in the contact element will be used for setting up the corresponding service

- The Watcher shall follow the procedures defined in section 6.2 "Default Watcher Processing" of [Presence2.0_DDS].

Regarding the use of the address provided in the contact, the communication addresses (contact) part of service tuples shall not be:

- Shown to the end-user, these addresses are handled locally by the terminal;
- Used to request presence subscription, an RCS client is NOT supposed to subscribe to the contact associated with a service capability tuple received in a presence document.

2.6.1.2.4 Authorisation for capabilities retrieval

To provide authorisation to retrieve the capabilities using an Anonymous Fetch request, an RCS client supporting the capability exchange using presence shall set the presence rules document in the presence XDMS as follows:

Presence XDMS:
AUID: org.openmobilealliance.pres-rules
Document name: pres-rules
Template

```xml
<?xml version="1.0" encoding="UTF-8"?>
<cr:ruleset
    xmlns:ocp="urn:oma:xml:xdm:common-policy"
    xmlns:op="urn:oma:xml:prs:pres-rules"
    xmlns:pr="urn:ietf:params:xml:ns:pres-rules"
    xmlns:cr="urn:ietf:params:xml:ns:common-policy">

    <! -- This rule allows all service capabilities to be sent for anonymous requests -->
    <! -- To realize the service capabilities to all requirement -->
    <! -- This rule replaces the default “wp_prs_block_anonymous” rule -->
    <! -- NOTE: May be modified to only allow RCS specified services -->
    <cr:rule id="rcs_allow_services_anonymous">
        <cr:conditions>
          <ocp:anonymous-request/>
        </cr:conditions>
        <cr:actions>
          <pr:sub-handling>allow</pr:sub-handling>
        </cr:actions>
        <cr:transformations>
          <pr:provide-services>
              <pr:all-services/>
          </pr:provide-services>
          <pr:provide-all-attributes/>
        </cr:transformations>
    </cr:rule>
</cr:ruleset>
```

Table 18: Presence XDMS template

If social presence is supported (see section 3.7), the *pres-rules* document should be set to contain both “*rcs_allow_services_anonymous*” described in this section and the rules provided in the template described in section 3.7.4.5.2.

Handling of this template shall be done as described in section 2.14.2.

2.6.1.2.5 Service part of the presence Data Model

A service capability is provided according to the model described in Table 19:

Attribute	Specification	Comment
entity	[RFC3863]	The entity field should be populated with a tel URI provided that the device has received a tel URI in P-Associated-URI header of 200 OK response to REGISTER request.
Tuple: <presence> -> <tuple>	[RFC3863] and [Presence2.0_DDS]	According to the presence schema defined in the [Presence], services are presented with *tuple* elements.
Status <tuple> -> <status> -> <basic> -> Open	[RFC3863] and [Presence2.0_DDS]	Mandatory element in [RFC3863]. Once a tuple element is published the value ‘open’ will always be used. It does not have any particular meaning in RCS context.
Service-id <tuple> -> <service-description> -> <service-id>	[Presence2.0_DDS]	*Service-description* element identifies a service and is described by a *service-id* and *version*. *Service-id* element contains a string that identifies a single service.

Attribute	Specification	Comment
Version <tuple> -> <service-description> -> <version>	[Presence2.0_DDS]	*Version* element contains the version number for the service, to identify different versions of the service (for example version number for specification number).
Media <tuple> -> <servcaps>	[RFC5196] and [Presence2.0_DDS]	Indicates the capabilities of the service. In RCS this is only used to provide media capabilities for some specific services (where mentioned below)
Contact <tuple> -> <contact>	[RFC3863] and [Presence2.0_DDS]	Contact element contains Presentity's communication address for the service. Contact address can be for example a tel or SIP URI, depending on the service used. The use of the Contact element is optional (if used it has to be a global routable URI) since the watcher may use the URI stored in the address book when initiating communication with the presentity. RCS Presentities either do not insert any contact element or insert a contact element for which the address matches the one used for identifying itself in communication (see Section 2.5) NOTE1: According to [RFC3863], "tuples that contain a *<basic>* element SHOULD contain a *<contact>* address". NOTE2: Populating *<contact>* element with GRUU may result in unpredictable watcher behaviour (see section 2.6.1.2.3.1) so GRUU should not be used Therefore, as a default- the *<contact>* element should be populated with a tel URI provided that: The device has received a tel URI in P-Associated-URI header of 200 OK response to REGISTER request. The service in question can utilise tel URIs.
Timestamp: <tuple> -> <timestamp>	[RFC3863] and [Presence2.0_DDS]	Timestamp when the presence information was published.

Table 19: Attributes of the Presence Service element

2.6.1.2.5.1 Service-descriptions for the Selected RCS Services

Service capabilities publication through OMA Presence Enabler [Presence2.0_TS] or [Presence] must follow [PDE_13] rules.

The RCS registered Service-description values are listed in OMNA Presence <service-description> Registry at:

http://www.openmobilealliance.org/Tech/omna/omna-prs-PidfSvcDesc-registry.aspx

The following *<service-description>* child elements, will be used in the presence document:

Standalone Messaging
Service-id: *org.openmobilealliance:StandaloneMsg*

Version: 2
Contact address type: tel / SIP URI

Session Mode Messaging
Service-id: *org.openmobilealliance:IM-session*
Version: 1.0
Contact address type: tel / SIP URI

Or

Service-id: org.openmobilealliance:ChatSession
Version: 2
Contact address type: tel / SIP URI

File Transfer via MSRP without Store and Forward
Service-id: *org.openmobilealliance:File-Transfer*
Version: 1.0
Contact address type: tel / SIP URI

This capability shall only be provided when File Transfer via MSRP is enabled through the PROVIDE FT configuration parameter described in section A.1.5 and the Chat Technology is set to OMA SIMPLE IM using the CHAT MESSAGING TECHNOLOGY configuration parameter defined in section A.1.4.3.

File Transfer with Store and Forward
Service-id: *org.openmobilealliance:File-Transfer*
Version: 2
Contact address type: tel / SIP URI

This capability shall only be provided when File Transfer via MSRP is enabled through the PROVIDE FT configuration parameter described in section A.1.5 and the Chat Technology is set to OMA CPM using the CHAT MESSAGING TECHNOLOGY configuration parameter defined in section A.1.4.3.

File Transfer Thumbnail
Service-id: *org.openmobilealliance:File-Transfer-thumb*
Version: 2
Contact address type: tel / SIP URI

This capability shall be provided when File Transfer via MSRP is enabled through the PROVIDE FT configuration parameter described in section A.1.5.

File Transfer via HTTP
Service-id: *org.openmobilealliance:File-Transfer-HTTP*
Version: 1.0
Contact address type: tel / SIP URI

Image Share
Service-id: org.gsma.imageshare
Version: 1.0
Contact address type: tel / SIP URI

Video Share during a call
Service-id: *org.gsma.videoshare*
Version: 1.0
Contact address type: tel / SIP URI

Social presence information
Service-id: *org.3gpp.urn:urn-7:3gpp-application.ims.iari.rcse.sp*
Version: 1.0
Contact address type: tel / SIP URI

Capability discovery via presence
Service-id: *org.3gpp.urn:urn-7:3gpp-application.ims.iari.rcse.dp*
Version: 1.0
Contact address type: tel / SIP URI

IP Voice Call (IR.92)
Service-id: *org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel*
Version: 1.0
Media capabilities: audio, duplex
Contact address type: tel / SIP URI

IP Video Call (IR.94)
Service-id: *org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel*
Version: 1.0
Media capabilities: audio, video, duplex
Contact address type: tel/ SIP URI

NOTE1: A single device supporting both IP Voice Call (IR.92) and IP Video Call (IR.94) shall publish a single tuple containing the common MMTel service id and both audio and video servcaps elements. Separate tuples are not required.

RCS IP Voice Call (strong preference for no breakout (IP end to end))
Service-id: *org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel.gsma.ipcall*
Version: 1.0
Media capabilities: audio, duplex
Contact address type: tel / SIP URI

RCS IP Video Call (strong preference for no breakout (IP end to end))
Service-id: *org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel.gsma.ipcall*
Version: 1.0
Media capabilities: audio, video, duplex
Contact address type: tel/ SIP URI

NOTE2: A single device supporting both RCS IP Voice Call and RCS IP Video Call with a strong preference for no breakout shall publish a single tuple containing the org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel.gsma.ipcall service-id and both audio and video servcaps elements. Separate tuples are not required.

RCS IP Video Call (strong preference for no breakout (IP end to end) and video media cannot be dropped)
Service-id: *org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel.gsma.ipcall.ipvideocallonly*
Version: 1.0
Media capabilities: audio, video, duplex
Contact address type: tel/ SIP URI

NOTE3: A device shall provide in the published capabilities only one of the RCS IP Voice Call, RCS IP Video Call and RCS IP Video Call where video media cannot be removed capabilities depending on whether according to the PROVIDE RCS IP VOICE CALL and PROVIDE RCS IP VIDEO CALL configuration parameters defined in section A.1.12 in the currently used radio technology respectively only RCS IP Voice Calls, both RCS IP Voice Calls and RCS IP Video Calls or only RCS IP Video Calls are allowed. Due the combining of the capabilities of different devices (i.e. the Presence Composition policy), it may happen though that multiple of those capabilities are received by a watcher. If the RCS IP Video Call where video media cannot be removed capability is received in combination with the RCS IP Video Call and/or the RCS IP Voice Call capability, towards that contact both RCS IP Voice Call and RCS IP Video Calls may be successfully initiated as described in sections 3.8 and 3.9 when those services are allowed according to the local configuration and used radio technology.

NOTE4: RCS IP Calls and IP Calls per IR.92/IR.94 can technically interoperate. Therefore, an RCS IP Call client is allowed to answer an incoming IP Call (and vice versa), and shall process the received capability information accordingly.

Geolocation PUSH
Service-id: *org.3gpp.urn:urn-7:3gpp-application.ims.iari.rcs.geopush*
Version: 1.0
Contact address type: tel/ SIP URI

Geolocation PULL
Service-id: *org.3gpp.urn:urn-7:3gpp-application.ims.iari.rcs.geopullft*
Version: 1.0
Contact address type: tel/ SIP URI

NOTE5: An RCS client shall include both the Video Share 2.0 and the Video Share 1.0 capabilities to indicate backwards compatibility with earlier RCS clients.

Call composer
Service-id: *org.3gpp.urn:urn-7:3gpp-service.ims.icsi.gsma.callcomposer*
Version: 1.0
Contact address type: tel / SIP URI

Post-call
Service-id: *org.3gpp.urn:urn-7:3gpp-service.ims.icsi.gsma.callunanswered*
Version: 1.0
Contact address type: tel / SIP URI

Shared Map
Service-id: *org.3gpp.urn:urn-7:3gpp-service.ims.icsi.gsma.sharedmap*
Version: 1.0
Contact address type: tel / SIP URI

Shared Sketch
Service-id: *org.3gpp.urn:urn-7:3gpp-service.ims.icsi.gsma.sharedsketch*
Version: 1.0
Contact address type: tel / SIP URI

The service capability information that is the object of a SIP PUBLISH by the RCS client (service tuple) corresponds to the services supported by the device. For example, a device can indicate its support for RCS IP Voice Calls according to section 3.8 with a service- and media description.

The set of services published may be further restricted by some Service Provider settings on the User Equipment (UE, on for example the services that are allowed by the Service Provider in the network) that are described in Annex A.

2.6.1.2.6 Future extensions to the mechanism

Consistently with section 2.6.1.1.3, it is also possible to extend the capability discovery based in presence following the guideline presented in the table below to define new service-IDs:

RCS service	Tag
Service Provider specific service (based on IARI)	Service-Id: org.3gpp.urn:urn-7:3gpp-application.ims.iari.rcs.mnc<mnc>.mcc<mcc>.<service name> Version: Service Provider choice
Service Provider specific service (based on OMA scheme)	Service-Id: org.openmobilealliance:<RCS service name>.mnc<mnc>.mcc<mcc>.<service extension> Version: Service Provider choice
Third-Party specific service	Service-Id: org.3gpp.urn:urn-7:3gpp-application.ims.iari.rcs. ext.<identifier>"

Table 20: Presence service tuple proposal for future lines of work

Service extension patterns including “mnc.mcc” may be registered with OMNA, if a service provider wishes to reserve the values in order to avoid any future collisions with new services (extensions, or new OMA services).

Example of the reserved service extension patterns that may be registered is:

org.openmobilealliance:ChatSession.mnc072.mcc01

Examples of service extensions:

- Service-id extension(s) for Group Chat with store and forward:
  org.openmobilealliance:ChatSession.mnc072.mcc01
  OR

org.openmobilealliance:ChatSession.mnc072.mcc01.myGCFlavor1 AND org.openmobilealliance:ChatSession.mnc072.mcc01.myGCFlavor2

- Using IARI:
- org.3gpp.urn:urn-7:3gpp-application.ims.iari.rcs.mnc01.mcc072.sfgroupchatMyFlavor

The use of the Third-Party specific services requires a full functioning framework, including a secure management of the Extensions on the devices. If this framework is not in place, Clients shall not enable third parties to use this service.

The Third-Party specific services are only exchanged in capability discovery and updates when the Extensions make use of the RCS infrastructure to communicate.

2.6.1.3 Coexistence between the discovery mechanisms

2.6.1.3.1 Service/capability indicators

The equivalence between presence Service-IDs and SIP OPTIONS tags are presented in the following table:

RCS service		Tag
Standalone Messaging	Tag	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.msg,urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.largemsg"
	Service Tuple	Service-id: org.openmobilealliance:StandaloneMsg Version: 2
Chat	Tag	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcse.im"
	Service Tuple	Service-id: org.openmobilealliance:IM-session Version : 1.0 Or Service-id: org.openmobilealliance:ChatSession Version : 2
File Transfer via MSRP (SIMPLE IM)	Tag	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcse.ft,urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.ftthumb"
	Service Tuples	Service-id: org.openmobilealliance:File-Transfer Version : 1.0 Service-id: org.openmobilealliance:File-Transfer-thumb Version : 2
File Transfer via MSRP (CPM)	Tag	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcse.ft,urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.ftthumb,urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.ftstandfw"
	Service Tuples	Service-id: org.openmobilealliance:File-Transfer Version : 2 Service-id: org.openmobilealliance:File-Transfer-thumb Version : 2

RCS service		Tag
File Transfer via HTTP	Tag	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.fthttp"
	Service Tuple	Service-id: org.openmobilealliance:File-Transfer-HTTP Version : 1.0
Image Share	Tag	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.gsma-is"
	Service Tuple	Service-id: org.gsma.imageshare Version: 1.0
Video Share during a call	Tag	+g.3gpp.cs-voice
	Service Tuple	Service-id: org.gsma.videoshare Version: 1.0
Social presence information	Tag	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcse.sp"
	Service Tuple	Service-id: org.3gpp.urn:urn-7:3gpp-application.ims.iari.rcse.sp Version: 1.0
Capability discovery via presence	Tag	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcse.dp"
	Service Tuple	Service-id: org.3gpp.urn:urn-7:3gpp-application.ims.iari.rcse.dp Version: 1.0
IP voice call (IR.92/IR.51)	Tag	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.mmtel"
	Service Tuple	Service-id: org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel Version: 1.0 Media capabilities: audio, duplex
RCS IP voice call	Tag	+g.gsma.rcs.ipcall
	Service Tuple	Service-id: org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel.gsma.ipcall Version: 1.0 Media capabilities: audio, duplex
IP video call (IR.94)	Tag	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.mmtel";video
	Service Tuple	Service-id: org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel Version: 1.0 Media capabilities: audio, video, duplex
RCS IP video call	Tag	+g.gsma.rcs.ipcall;video
	Service Tuple	Service-id: org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel.gsma.ipcall Version: 1.0 Media capabilities: audio, video, duplex

RCS service		Tag
	Tag (RCS IP video call where video media cannot be removed)	+g.gsma.rcs.ipvideocallonly
	Service Tuple (RCS IP video call where video media cannot be removed)	Service-id: org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel.gsma.ipcall.ipvideocallonly Version: 1.0 Media capabilities: audio, video, duplex
Geolocation PULL	Tag	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.geopullft"
	Service Tuple	Service-id: org.3gpp.urn:urn-7:3gpp-application.ims.iari.rcs.geopullft Version: 1.0
Geolocation PUSH	Tag	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.geopush"
	Service Tuple	Service-id: org.3gpp.urn:urn-7:3gpp-application.ims.iari.rcs.geopush Version: 1.0
Call composer	Tag	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service. ims.icsi.gsma.callcomposer"
	Service Tuple	Service-id: org.3gpp.urn:urn-7:3gpp-service.ims.icsi.gsma.callcomposer Version: 1.0
Post-Call	Tag	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service. Ims.icsi.gsma.callunanswered"
	Service Tuple	Service-id: org.3gpp.urn:urn-7:3gpp-service.ims.icsi.gsma.callunanswered Version: 1.0
Shared Map	Tag	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service. ims.icsi.gsma.sharedmap"
	Service Tuple	Service-id: org.3gpp.urn:urn-7:3gpp-service.ims.icsi.gsma.sharedmap Version: 1.0
Shared Sketch	Tag	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service. ims.icsi.gsma.sharedsketch"
	Service Tuple	Service-id: org.3gpp.urn:urn-7:3gpp-service.ims.icsi.gsma.sharedsketch Version: 1.0

Table 21: Complete SIP OPTIONS tag and Presence Service ID usage for RCS

2.6.1.3.2 Coexistence using a common device stack

As mentioned in section 2.6.1, the principle for interoperability is to have a common stack on devices which is able to:

- Answer a SIP OPTIONS query as per the mechanism presented in section 2.6.1.1 independently on whether the device is configured to use SIP OPTIONS or presence as the default capability exchange mechanism.
- If the device is configured to use presence as the default capability exchange mechanism, implement the fallback to SIP OPTIONS procedure.

2.6.1.3.2.1 Interworking when the request is originated in the Service Provider using presence as the default discovery mechanism

In this case, the initial capability exchange request is performed using presence (ANONYMOUS SUBSCRIBE), however either the originating or the terminating Service Provider Network detects that this method is not supported for that particular user and returns with one of the following errors:

- 405 METHOD NOT ALLOWED
- 501 NOT IMPLEMENTED

As a result, the RCS stack on the UE shall identify that the contact does not support the Presence discovery mechanism. From thereon the OPTIONS-based mechanism (as presented in section 2.6.1.1) shall be used to query that contact’s capabilities.

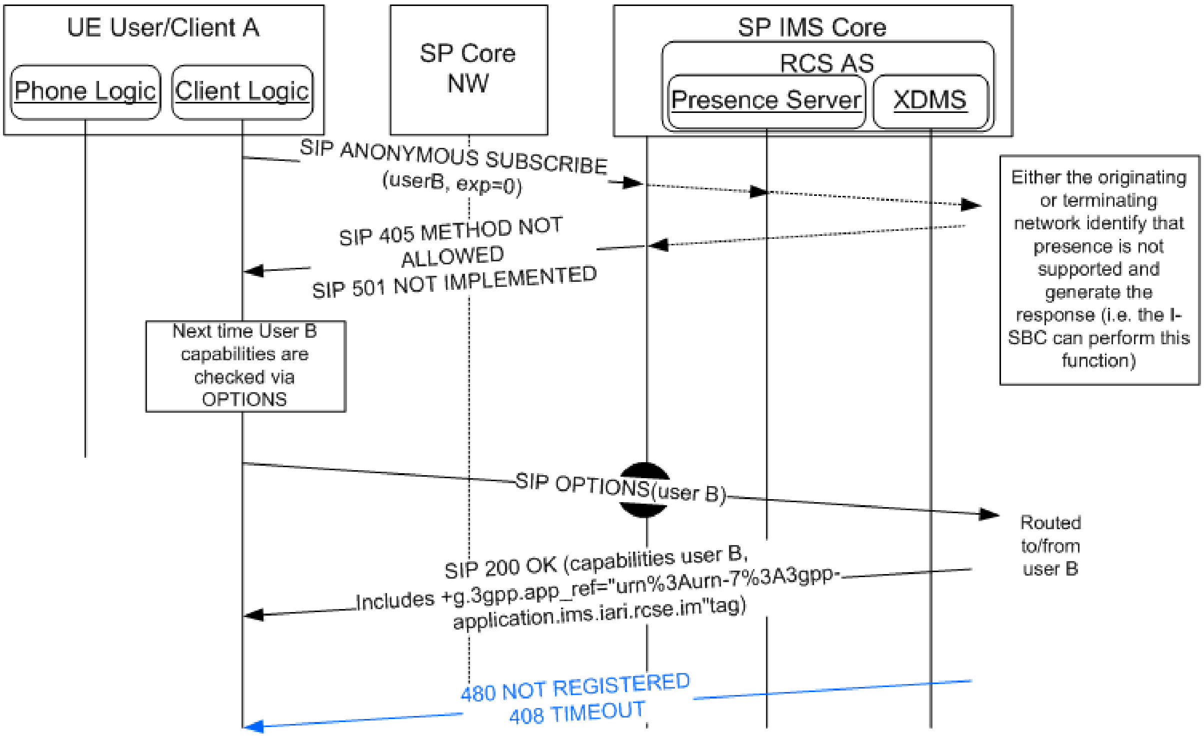

Figure 8: Fallback to SIP-OPTIONS procedure

If in the future, the contact is again identified as supporting discovery via presence (i.e. the *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.rcse.dp"* tag was included either in the OPTIONS request or in its response), then capability discovery via presence (as described in section 2.6.1.2) will be used from there on for that contact.

2.6.1.3.2.2 Interworking when the request is originated in the Service Provider using SIP OPTIONS as the default discovery mechanism

In this case, the SIP OPTIONS message is exchanged end-to-end as described in section 2.6.1.1.1.

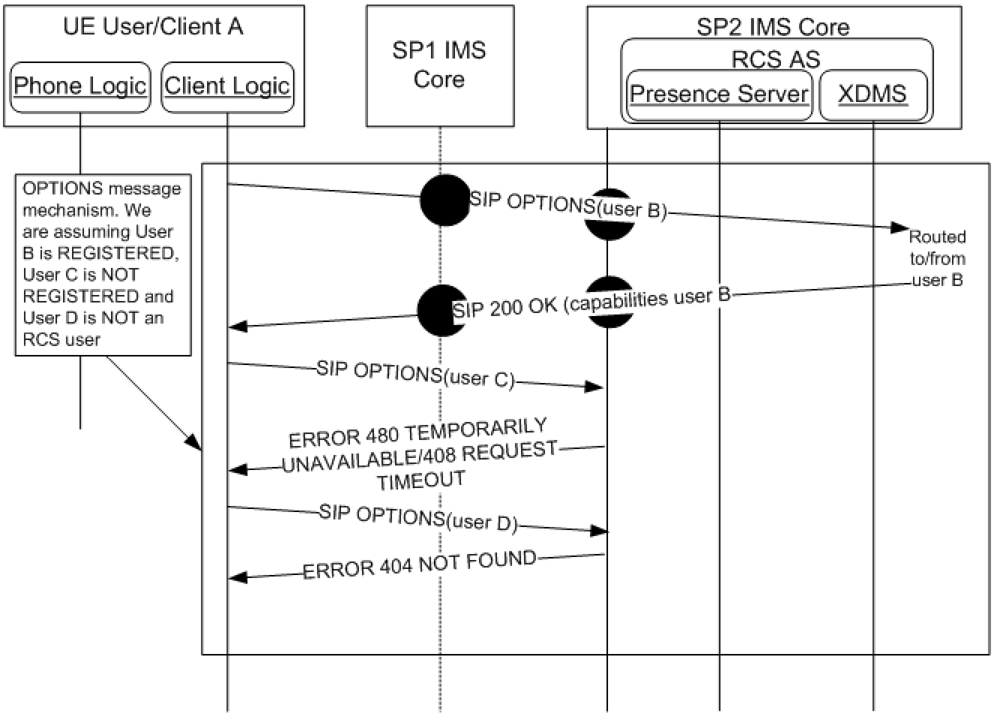

Figure 9: Inter-Service Provider SIP OPTIONS exchange for interworking

2.6.1.3.3 Coexistence between the discovery mechanisms via network interworking

When Service Providers use presence as the discovery mechanism, there are two ways in which interoperability is achieved between such a Service Provider and those Service Providers who have selected SIP OPTIONS as the default discovery mechanism.

- Service Provider supports fallback to SIP OPTIONS: Interoperability leverages the common device stack as defined in 2.6.1.3.2 above. In this case, there is no requirement for network based interworking.
- Service Provider does not support fallback to SIP OPTIONS: Interoperability is provided by network based interworking.
- Service Provider disables capability discovery: Interoperability is provided by network based interworking.

Refer to the interworking table below to identify specific network based interworking requirements:

			Service Provider A			
			Default: SIP OPTIONS		Default: Presence	
			No Presence Server	Presence Server	OPTIONS Fallback	No OPTIONS fallback
Service Provider B	Default: SIP OPTIONS	No Presence Server	No Network Interworking Required[2]	No Network Interworking Required[2]	No Network Interworking Required[1]	**Bidirectional Interworking required[3]**
		Presence Server	No Network Interworking Required[2]	No Network Interworking Required[2]	No Network Interworking Required[1]	**Unidirectional Interworking required[4]**
	Default: Presence	OPTIONS Fallback	No Network Interworking Required[1]	No Network Interworking Required[1]	No Network Interworking Required[2]	No Network Interworking Required[2]
		No OPTIONS fallback	**Bidirectional Interworking required[3]**	**Unidirectional Interworking required[4]**	No Network Interworking Required[2]	No Network Interworking Required[2]

Table 22: Service Discovery network-based Interworking summary

NOTES:
1. No interworking required; based on common stack approach.
2. No interworking required; based on common default discovery mode.
3. Interworking required for SIP OPTIONS conversion to SUBSCRIBE/NOTIFY and vice versa.
4. Interworking required for SIP OPTIONS conversion to SUBSCRIBE/NOTIFY; requirement for conversion from SUBSCRIBE/NOTIFY to SIP OPTIONS contingent upon "SIP OPTIONS default" Service Provider support for Anonymous Fetch at PS.

Table 22 considers whether a service provider that uses SIP OPTIONS as the default discovery also supports presence or not:

- The Presence Server acts as a source for both SPI and capability information. This is addressed in 2.6.1.4 which states that capability exchanges are not required in the case where a social relationship is established.
- If a Service Provider uses SIP OPTIONS as the default discovery mechanism, and has deployed presence, the Service Provider may implement a policy that allows their Presence Server to respond to presence based discovery (anonymous) requests.
- Such a policy would impact the required interworking architecture; therefore it is addressed in Table 22 above.

Specific network interworking function requirements are contingent upon the service discovery modes and policies of each service provider. At the Service Provider's discretion, an interworking function can be implemented in the network to:

- Answer incoming SIP OPTIONS requests based on the Presence Server information (Figure 10).
- Convert SIP ANONYMOUS SUBSCRIBE requests into SIP OPTIONS requests (Figure 11).

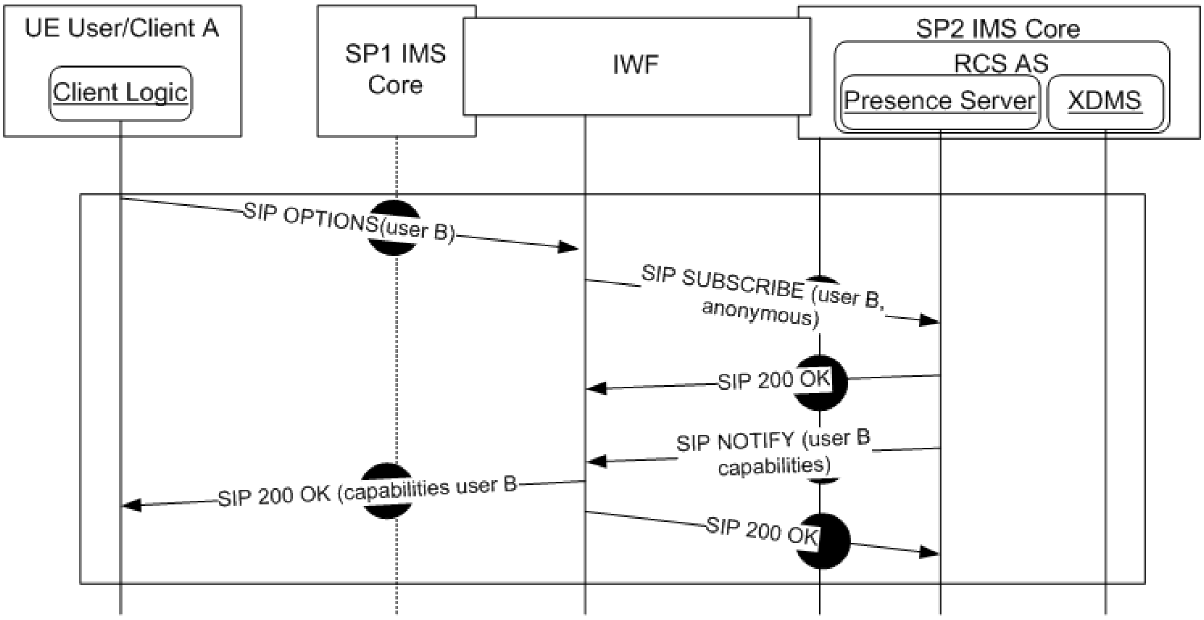

Figure 10: Capability interworking via network: Options request

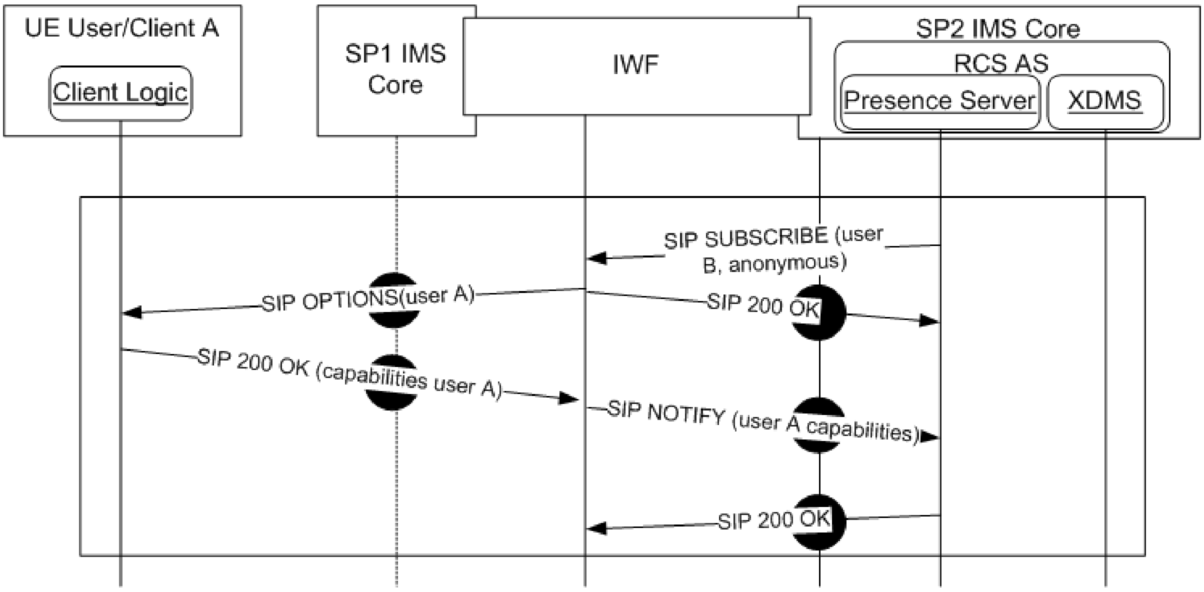

Figure 11: Capability interworking via network: Presence request

NOTE: Figure 10 and Figure 11 do not specify whether the IWF is deployed:

- In the Originating IMS network or
- In the Terminating IMS network or
- In the Inter-Network region.

All of the above are valid architectural options. NNI impact is not uniform and is a function of the architecture selected. While the details surrounding the specific architecture and functionality of an IWF are left to the Service Provider, it is recommended that impact at the UNI should be minimal and as transparent as possible.

The successful deployment of network IWF capabilities must provide an environment where all RCS devices exchange capabilities information without requiring additional functionality or logic at the client (i.e. no UNI impact).

The following additional guidelines are provided regarding the implementation of an IWF function:

- If either Service Provider has a heterogeneous network from a capabilities discovery mode perspective, this must be factored into the IWF architecture.
- The Service Provider implementing an IWF must consider policy aspects of the functionality. This includes any decisions to filter or transform service capabilities across the IWF.
    - Domain/Service Provider based policies; i.e. specific services are configured to be exposed based on the destination domain.
    - Service level policies: specific services, including Service Provider proprietary or other specialised services that may be filtered from exposure to any external domains.
    - User based policy; including privacy or other subscriber level policies.

2.6.1.4 Capability discovery and social presence information coexistence

In the following two cases:

- The mechanism for capability discovery is performed via SIP OPTIONS and the Service Provider has decided to deploy a Presence Server to provide the SPI service[6].
- The discovery mechanism is based on presence

Then for those contacts who have a social presence relationship established with the sender, it is not necessary to perform a capability exchange because their capabilities will be updated automatically using the standard SPI mechanisms described in section 3.7.4.

2.6.1.5 Capability exchange optimisations

To avoid the overhead and increase the efficiency, the client may implement optimisation mechanisms as listed in section 2.6.4.

2.6.2 User discovery mechanism

With the main aim of optimising, the UX and minimising the unnecessary traffic generated by an RCS client, a set of lists shall be generated and maintained by the UE or client:

- A list of the RCS enabled users as the list of users which support at least one RCS service and obviously the capability discovery framework. It should be noted that, the first view of the address book shall use this list to clearly identify the RCS capable contacts with a visual RCS flag.
- One individual list per RCS service of RCS contacts which are enabled to perform that particular service.

These lists should include both registered and non-registered contacts; in contrast, it does not include non-provisioned contacts.

To keep these lists up-to-date, the UE or client shall use one of the capability discovery mechanisms presented in section 2.6.1 in the following scenarios:

- When a new contact is added to the phonebook. The new contact may come from different sources and, therefore, the mechanism described in the following sections applies to all the scenarios presented below:

    - Contact added manually by the user.
    - Synchronised via 3rd party servers or PC.
    - Received via Bluetooth or handling a vCard file received, for example via e-mail.

---

[6] It may be possible for a Service Provider to always perform service discovery via SIP OPTIONS, but have a policy allowing for remote domain (NNI) support for discovery via presence as discussed in 2.6.1.3.3. This would allow a remote Service Provider that does not support fallback to SIP OPTIONS to obtain capability information using anonymous SUBSCRIBE without traversing a network IWF.

- The first time the user accesses the service from a new device, the whole address book needs to be polled if not disabled through the DISABLE INITIAL ADDRESS BOOK SCAN client configuration parameter (Annex A section A.1.10 for reference).
- Periodically (frequency determined by the POLLING PERIOD parameter described in Annex A section A.1.10) to all the contacts in the phone address book whose capabilities are not available (e.g. non-RCS users) or are expired (see CAPABILITY INFO EXPIRY parameter in Annex A section A.1.10 for reference).
- When a contact's details are edited thereby modifying the information which is used to identify the contact as RCS (as described further in section 2.6.2 e.g. the MSISDN is modified or a new MSISDN is added).

Additionally, it should be noted that if a client is NOT registered at the time the new contact(s) are added, the client should keep the necessary information on the device. In this case the capabilities shall be verified the next time the RCS client completes the registration process.

2.6.2.1 Discovery via OPTIONS message

The SIP OPTIONS message can be employed not only to determine the capabilities but also to identify whether or not a contact is an RCS user; independently from whether the contact is registered at the time the query is performed.

When a SIP OPTIONS message is sent from User A to User B, User A will learn about user B's capabilities through one of 6 scenarios:

1. If User B is registered and User A is not in the capability blacklist of User B, then the response from User B's client will include the CAPABILITY STATUS – the set of services currently available (based on tags as described in section 2.6.1.1.2), else if User B is registered and User A is in its blacklist, it will only answer User A with an empty 200 OK. Please note that regarding the list of RCS users, the contact shall be only considered as an RCS user, if the response (SIP 200 OK) includes any of the tags described in Table 21.
2. If User B is currently not registered (e.g. the device is switched off, out of coverage or roaming with data services disabled), then the network will respond with one of the following error messages: SIP 480 TEMPORARILY UNAVAILABLE (graceful deregistration took place) or SIP 408 REQUEST TIMEOUT. From the new user discovery point of view, this response is ignored because it is inconclusive:

    - It does not confirm whether the contact is an RCS user, and,
    - It does not provide any relevant update to the list of RCS contacts capable of a particular service

3. If User B is not provisioned for RCS the network will respond with a message error: SIP 404 NOT FOUND[7]. Therefore, if this message is received, the user is identified

[7] Please note that the response provided may depend on the network configuration. A useful approach for the terminal is to parse the response and if it is not either a 200 OK containing the capabilities as feature tags, a 480 TEMPORARILY UNAVAILABLE or a 408 REQUEST TIMEOUT, the target user should be considered as non-RCS. For simplicity, the present document assumes in

as a non-RCS user (removed from the list of RCS users and from the individual list of RCS users capable of a particular service)

4. If User B was previously identified as an RCS user and the response to the OPTIONS message indicates that User B is no longer supporting any RCS services, User B should be identified as a non-RCS user and, consequently, removed from the list of RCS enabled contacts
5. In addition to this and based on the fact the SIP OPTIONS request contain the list of services supported by the requester, the receiver shall use the SIP OPTIONS message to update both the RCS contact list and the relevant per service lists as per the criteria presented in the previous four scenarios .
6. Please note there is a possibility an RCS user who is not within the address book contacts may send OPTIONS messages or responses (e.g. when receiving a call or making a call using a MSISDN not included in the contacts). In this case the capabilities shall be stored temporarily (at least 20 seconds from when OPTIONS is received) in the terminal to:

   - Keep the service availability updated while a session (Chat, File Transfer, Video/Image Share, IP Voice or Video call, Geolocation PUSH) is still in place, and,
   - To add the information to the new contact (both the fact that it is an RCS user and the cached capabilities) if the user decides to add a new address book entry following a communication.

To illustrate the behaviour, the following example is provided. User A is registered and decides to add or modify a new contact which results in a new IMS identity for the contact (e.g. new MSISDN which implies a new tel URI). As a consequence, the client is required to verify whether the contact is an RCS user and therefore, add them to the list the terminal maintains.

The following sections that the response provided by the Service Provider core network is always 404 NOT FOUND, however, the previous statement should be taken into account.

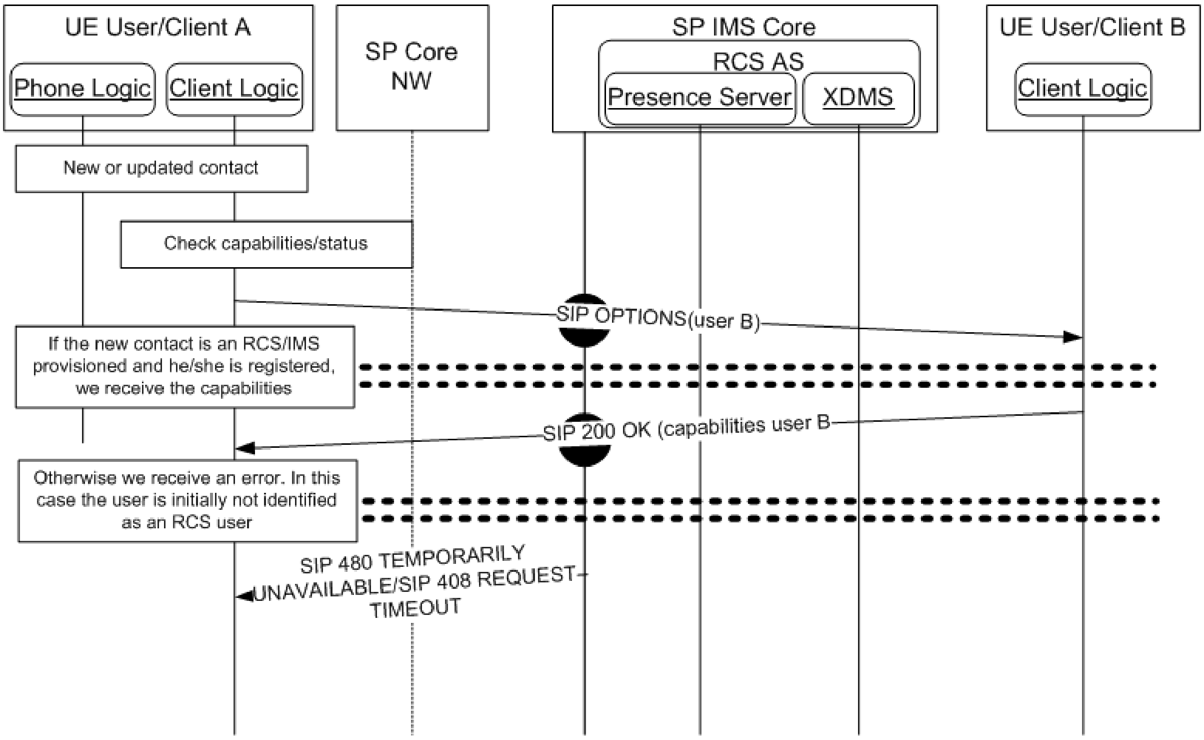

Figure 12: Adding/Editing a contact

2.6.2.2 Discovery via PRESENCE

The procedure for user discovery using presence is analogous to the capability discovery procedure using presence as described in section 2.6.1.2. However the following additional considerations shall be taken into account:

- When User A queries User B's capabilities, the response will include the CAPABILITY STATUS – the set of services currently available (based on the service-IDs presented in section 2.6.1.3.1). Please note that regarding the list of RCS users, the contact shall be considered as an RCS user, only if the response includes one of the service-IDs described in Table 21.

2.6.2.3 Coexistence between user discovery mechanisms

Please note that the mechanisms described in sections 2.6.1.3 also apply to the user discovery mechanisms co-existence.

2.6.2.4 User discovery and social presence information coexistence

Please note that the considerations presented in section 2.6.1.4 also apply to the user discovery process.

2.6.2.5 Capability polling mechanism

To enhance the discovery of new users and, ultimately, keep the list of RCS contacts up to date, this specification provides a mechanism, capability polling, consisting of the polling of the status/capabilities of all the contacts in the address book whose capabilities are not available (such as non-RCS users) or have expired (see CAPABILITY INFO EXPIRY parameter in Annex A section A.1.10 for further reference).

It should be noted that the capability polling mechanism is optional and will be only performed if the related configuration settings have been provisioned (that is if the POLLING PERIOD parameter presented in Annex A section A.1.10 is set to 0, this polling mechanism will not be used).

Assuming the POLLING PERIOD is configured to be greater than 0 and after the polling timer expires, the client will use the following mechanism to update the list of RCS contacts and update their capabilities.

Please note the capability polling is only performed on:

- Those contacts without capability information (non-RCS users and RCS users with unknown capabilities), and,
- The rest of RCS contacts provided the associated capability information is older than the CAPABILITY INFO EXPIRY parameter (see Annex A section A.1.10 for further reference)8.

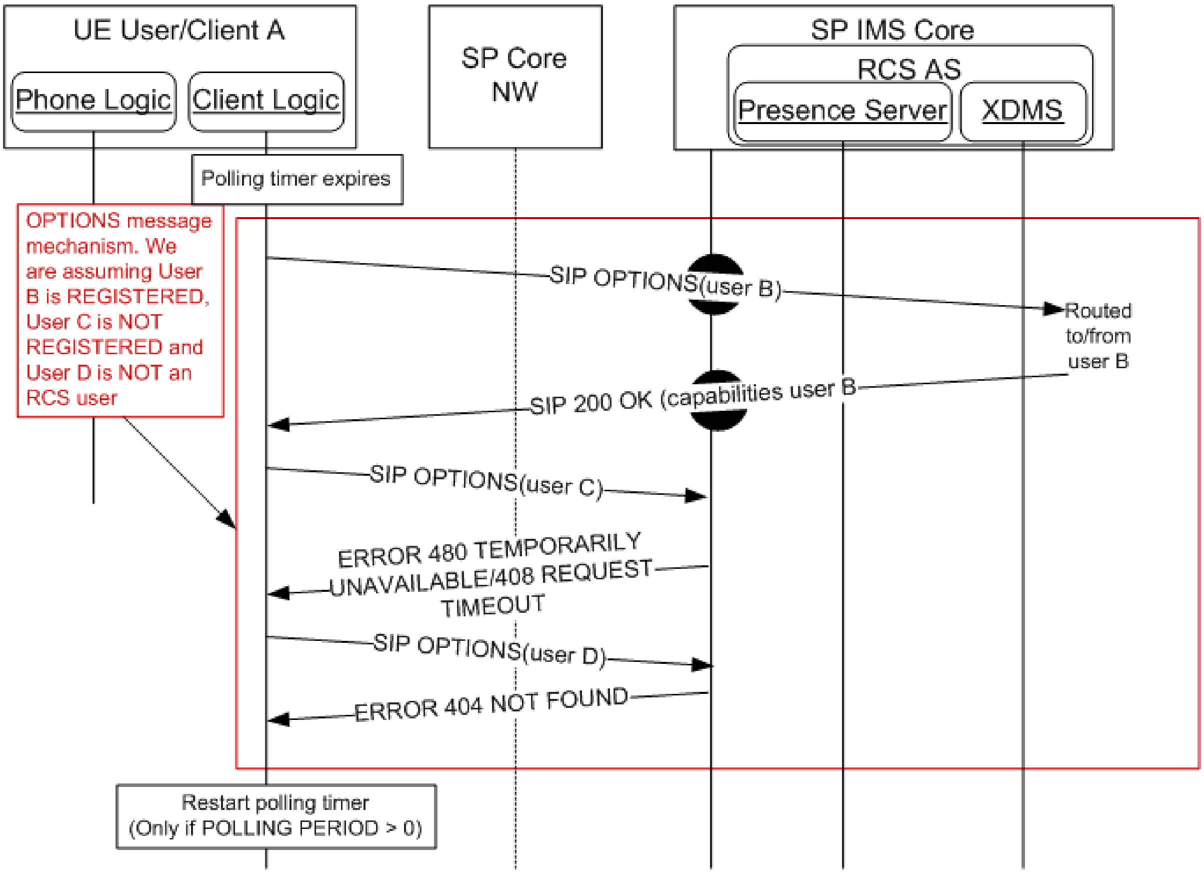

Figure 13: Capabilities polling via OPTIONS message

When CAPABILITY DISCOVERY MECHANISM is set to presence (see Annex A section A.1.10), the presence based discovery based in the use of SIP ANONYMOUS SUBSCRIBE requests are used for all the contacts except:

8 Please note this is a traffic optimization to reduce the amount of SIP OPTIONS messages generated by capability polling

- If implementing co-existence based on a common device stack, those contacts which are identified as not supporting presence discovery (SIP OPTIONS will be used instead as per the fallback procedure presented in section 2.6.1.3.2.1).
- Those users with a SPI relationship in place because their capabilities will be updated automatically using the standard SPI mechanisms described in section 3.7.4.

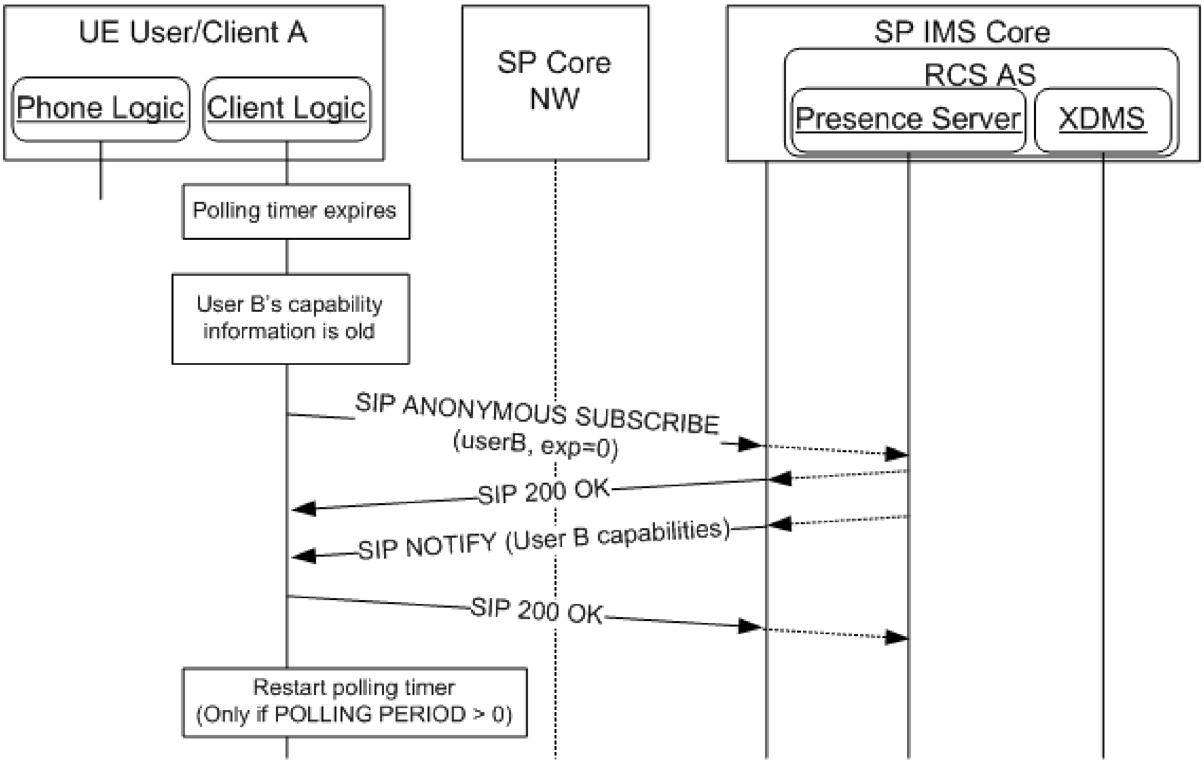

Figure 14: Capabilities polling via anonymous fetch

NOTE: If an RLS-URI was provisioned (see Annex A Section A.1.2.1) and the capabilities of multiple contacts need to be queried, the capability query could be initiated by the device using a request contained list that is decomposed by the RLS service in the originating network (see section 2.6.1.2.3 for more details). In this case the SIP SUBSCRIBE request shown in Figure 14 would be a back end subscribe issued by the user's home RLS and should be forwarded to the correct destination Presence Server(s). The RLS will gather the notifications and send aggregated notifications to the device.

Finally, and as a summary of the capability and new user discovery mechanism composition the following diagram is provided.

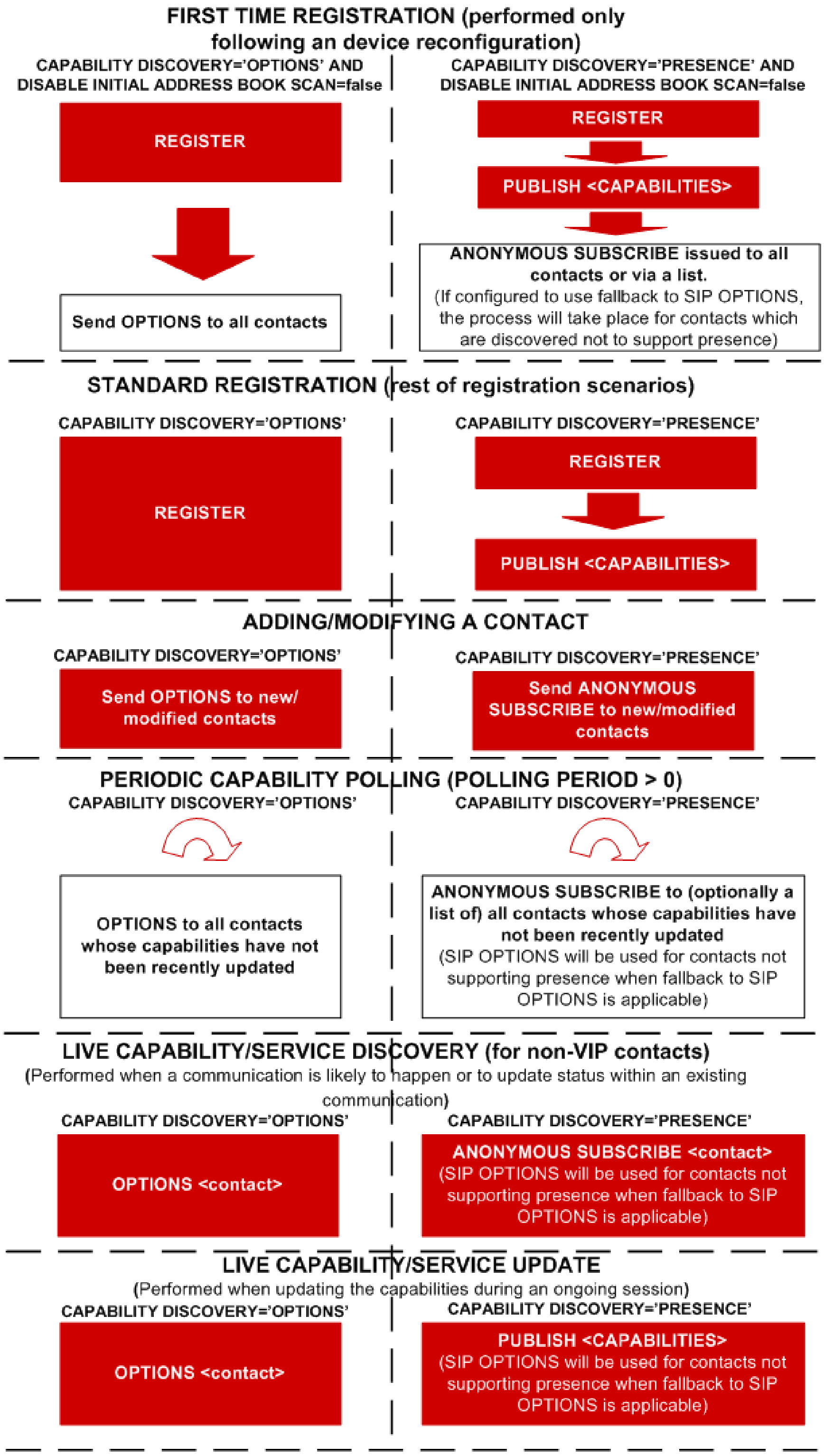

Figure 15: RCS capability and new user discovery mechanisms

The red boxes represent mandatory procedures. Meanwhile, the clear boxes represent optional procedures.

2.6.3 Capability update for services

A capability update shall only be triggered for contacts belonging to

- the list of the RCS enabled users as defined in section 2.6.2, or
- the non RCS contacts whose last capability check is older than the NON RCS CAPABILITY INFO EXPIRY configuration parameter value defined in section A.1.10.

2.6.3.1 Entry points for capability update

A capability update is triggered by one of the following activities:

- If not disabled[9], after first time registration to obtain the registration state and default set of capabilities for each contact in the device's address book (note one capability exchange takes place per IMS identity [that is tel URI/MSISDN or SIP URI] stored in the address book)[10],
- When checking the available RCS services/capabilities to communicate with another user (e.g. from the address book and call-log)
- After establishing voice call to obtain the real-time capabilities for the call or Chat session provided this has not been performed before (see previous bullet) or content sharing during a call is supported.
- After the call returns to an active state (e.g. returning from call wait, call on hold or multiparty call).
- When a communication is active with a user to provide an update when the relevant available capabilities change:

  - When a 1-2-1 Chat session is established and any of the following capabilities change:

    - File Transfer via MSRP
    - File Transfer via HTTP
    - Geolocation PUSH

  - When in an active call with an RCS user and any of the following capabilities change:

    - Chat
    - File Transfer via MSRP
    - File Transfer via HTTP
    - Geolocation PUSH
    - Geolocation PULL
    - Video Share
    - Image Share
    - IP Video Call
    - Shared Sketch

[9] Through the DISABLE INITIAL ADDRESS BOOK SCAN client configuration parameter

[10] Please note a contact may have several MSISDNs or associated SIP URIs. The client will use ALL the MSISDNs/SIP URIs stored for that user to perform the capability exchange. If it is discovered that more than one of the associated tel URIs/SIP URIs are IMS provisioned, each will be treated as a separate RCS user. For example, if displaying the list of RCS contacts, two or more entries for a user will be shown ("John Smith mobile" and "John Smith home"), so the user can choose.

    - Shared Map

  - When an IP call or video call session is in place and any of the following capabilities change:

    - Chat
    - File Transfer via MSRP
    - File Transfer via HTTP
    - Geolocation PUSH
    - Geolocation PULL
    - Video Share
    - Image Share
    - IP Video Call
    - Shared Sketch
    - Shared Map

- When there is a communications event (text, email, call or Chat) with another user in the address book, taking into account the optimisations presented in section 2.6.1.5.
- When performing one of the actions described in section 2.2 of [PRD-RCC.20].

Actions related to services described above apply only when the corresponding service is authorised by configuration.

2.6.3.1.1 UX guidelines: Access to RCS services through address book and call-log interaction

The address book (and by extension the call-log window as an alternative for users who have been recently phoned) is the centrepiece to access all RCS services. From it, the user is able to:

- Identify which services are available for each contact: When a contact is selected, the capabilities are updated using one of the mechanisms described in section 2.6 (SIP OPTIONS query or PRESENCE), and the result is presented to the user by showing the RCS services which are available to communicate with that particular contact

  - Please note for those contacts who have a social presence relationship established with the sender, it is not necessary to perform a capability exchange because their capabilities will be updated automatically using the standard SPI mechanisms described in section 3.7.4. Therefore and for those contacts, the capability exchange is not required.

- If one or more RCS services are available[11], they can be started from the address book/call log entry. Please note the only exception is for those content sharing

[11] It should be noted that in this case if IM CAP ALWAYS ON (see Table 75) is enabled, the Chat should still be reported to the user as available even if the other end/user is not registered. It may also be offered if the client is not registered itself. In that case the composed messages should be queued to be sent when the device comes online again. The messages should be sent as described in section 2.7.1.1.

services that can be only accessed when during a call. If a contact has more than one RCS capable telephone number assigned a device should either display for each of these individual numbers which services are available or for each RCS Service the individual telephone numbers on which it is available.

In addition to this, the first view of the address book may clearly identify the RCS capable contacts with an icon or flag.

2.6.3.1.1.1 General assumptions

The following sections describe the relevant chat message flows and reference UX. Please note that the following assumption has been made:

- For simplicity, the internal mobile network interactions are omitted in the diagrams shown in the following sections.

2.6.3.1.1.2 Capability update process

The capabilities update process is described in the following diagram. In this case the contact (User B) is an RCS contact which is registered.

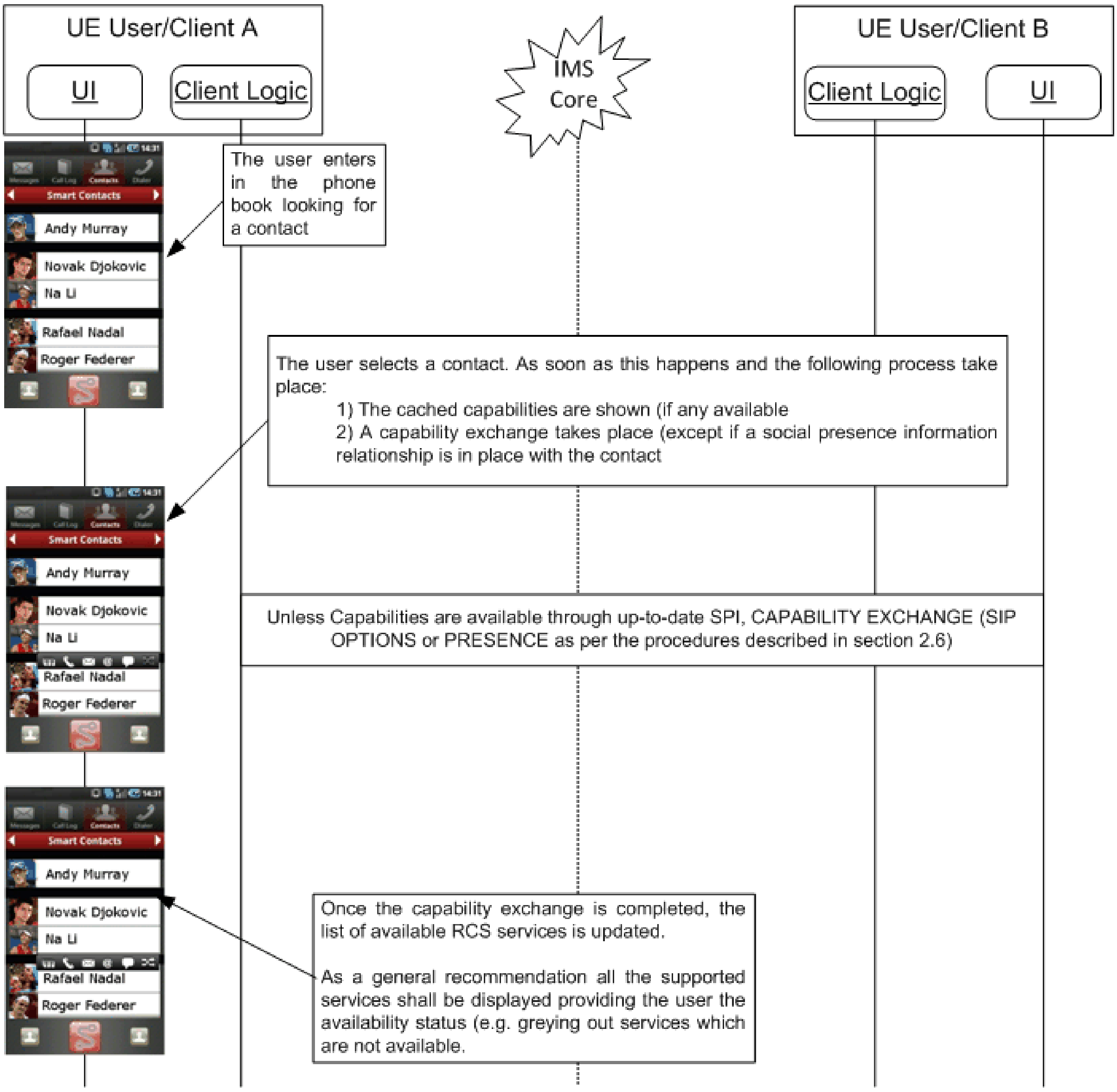

Figure 16 : Address book and call-log service access: Capabilities update

NOTE: If User B is either not an RCS user or they are not currently registered, User A's client may assume that no services are available to communicate with User B.

As a general recommendation, when capability discovery is enabled, all the supported RCS services should be displayed providing the user the availability status (e.g. greying out services which are not available).

2.6.3.2 Standalone messaging: Text and multimedia messaging

The capability exchange is not required for this service when network interworking to a fall-back technology is available.

2.6.3.3 1-to-1 Chat

In addition to the general user driven entry points and taking into account:

- The optimisations provided in section 2.6.1.5, and,

- The potential impact of having an SPI relationship between sender and receiver as described in section 2.6.2.4.

The capability exchange shall take place (when the service capability query is supported by SIP OPTIONS, PRESENCE or when the contact is not a VIP Contact for SPI):

- Before the initial SIP INVITE is sent to initiate the service to verify if the receiver is ready for the service (unless an exchange of capabilities has just been made based on one of the criteria listed in section 2.6.3.1, as described in section 2.6.4)
- If an error takes place:
    - After the Chat session is abruptly terminated or irregular signalling behaviour during the establishment of the service is detected.
    - When there is an update on the available capabilities on either end once the session is established.
- In any of the scenarios described in section 2.6.3.1 which are relevant to the service.

2.6.3.4 Group Chat

In addition to the general user driven entry points and taking into account:

- The optimisations provided in section 2.6.1.5, and,
- The potential impact of having an SPI relationship between sender and receiver as described in section 2.6.2.4.

The capability exchange shall take place (when the service capability query is supported by SIP OPTIONS, PRESENCE or when the contact is not a VIP Contact for SPI):

- Before the initial SIP INVITE is sent to initiate the service to verify if the receiver is ready for the service (unless an exchange of capabilities has just been made based on one of the criteria listed in section 2.6.3.1, as described in section 2.6.4)
- If an error takes place:
    - After the Chat session is abruptly terminated or irregular signalling behaviour during the establishment of the service is detected.
    - When selecting the participants of a Group Chat to verify whether they are available.

2.6.3.5 File Transfer

In addition to the general user driven entry points and taking into account:

- The optimisations provided in section 2.6.1.5, and,
- The potential impact of having an SPI relationship between sender and receiver as described in section 2.6.2.4.

The capability exchange shall take place (when the service capability query is supported by SIP OPTIONS, PRESENCE or when the contact is not a VIP Contact for SPI):

- Before the initial SIP INVITE is sent to initiate the service to verify if the receiver is ready for the service (unless an exchange of capabilities has just been made based on one of the criteria listed in section 2.6.3.1, as described in section 2.6.4)
- If an error takes place:
    - After the service is cancelled either by the sender or receiver.
    - After the file transfer is abnormally interrupted as a result of a failure or irregular signalling behaviour during the establishment of the service is detected.

2.6.3.6 Content sharing

In addition to the general user driven entry points and taking into account:

- The optimisations provided in section 2.6.1.5, and,
- The potential impact of having an SPI relationship between sender and receiver as described in section 2.6.2.4.

When enabled, the capability exchange shall take place:

- Before the initial SIP INVITE is sent to initiate the service to verify if the receiver is ready for the service (unless an exchange of capabilities has just been made based on one of the criteria listed in section 2.6.3.1, as described in section 2.6.4)
- If an error takes place:
    - After the service is cancelled either by the sender or receiver.
    - After the sharing is abnormally interrupted as a result of a failure or irregular signalling behaviour during the establishment of the service is detected.
    - After the call is no longer active.
- In any of the scenarios described in section 2.6.3.1 which are relevant to the service.

Additionally to the previous capabilities query entry point, a client provider may implement additional ones to enhance the user experience. For example, it may be considered to issue a capability exchange when the relevant sensors in a device indicate that the user is likely to interact with the phone keyboard or screen during a call.

2.6.3.7 Social presence

Information indicating support for social information via presence is expected prior to a user's attempt to establish a social presence relationship. This supports the "Who Can I Invite" concept; providing the user with a view of the contacts with whom they can attempt to establish a social presence relationship. This information is provided in the following contexts:

- Discovery via SIP OPTIONS.
- Discovery via Presence

Independently of the chosen mechanism,

- If capability discovery indicates that both clients support the "social information via presence" functionality, the user is then presented with the possibility of inviting the contact to share the social presence information. This includes invitation of a

previously discovered SPI-enabled contact who is temporarily Not Available. If not, the terminal should not present this possibility to the user for that contact.
- For those contacts who have an active social presence relationship established with the sender, it shall not perform a capability exchange if their capabilities are updated automatically using the standard SPI mechanisms described in section 3.7.4.

2.6.3.7.1 Discovery via SIP OPTIONS

To ensure interoperability[12] and enable those Service Providers implementing an SIP OPTIONS based capability/user discovery mechanism for their RCS deployments but deploying a Presence Server to additionally provide the social profile information (as described in section 3.7.4.2.2) functionality, the UE shall provide the following procedure:

- Prior to being able to send an invitation to a contact (e.g. from the address book), the terminal will use the OPTIONS mechanism to determine if the other end also supports this feature (that is both ends include the “Social Presence Information” SIP OPTIONS tag in the relevant headers).

2.6.3.7.2 Discovery via presence

Prior to being able to send an invitation to share Social Presence with a contact (e.g. from the address book), the terminal may use the Anonymous Fetch mechanism to determine if the other end also supports this feature (that is both ends include an “open” “Social Presence Information” Presence Service Tuple in the Presence Information Data Format [PIDF]). This includes inviting a contact who has previously been discovered to be Social presence-enabled even when they are currently offline.

2.6.3.8 IP Voice Call

2.6.3.8.1 IP Voice Call per MMTEL

The capability exchange is not required for this service. This capability may be used for network internal use and shall not have an impact on the user experience.

2.6.3.8.2 RCS IP Voice Call

In addition to the general user driven entry points and taking into account:

- The optimisations provided in section 2.6.1.5, and,
- The potential impact of having an SPI relationship between sender and receiver as described in section 2.6.2.4.

When enabled, the capability exchange shall take place:

- Before the RCS IP Voice Call is initiated by the sender to verify if the receiver is ready for the service (unless an exchange of capabilities has just been made based on one of the criteria listed in section 2.6.3.1, as described in section 2.6.4)
- If an error takes place, after the call when the service was abnormally interrupted or irregular signalling behaviour during the establishment of the call is detected.

---

[12] Please note that the present specification allows the deployment of RCS communication services without the need for a Presence Server and the associated XDM servers, therefore, the present specification provide the necessary guidance to secure interoperability.

- In any of the scenarios described in section 2.6.3.1 which are relevant to the service.

2.6.3.9 IP Video Call

2.6.3.9.1 IP Video Call per MMTEL

In addition to the general user driven entry points and taking into account:

- The optimisations provided in section 2.6.1.5, and,
- The potential impact of having an SPI relationship between sender and receiver as described in section 2.6.2.4.

When enabled, the capability exchange shall take place:

- Before the IP Video Call per MMTEL is initiated by the sender to verify if the receiver is ready for the service (unless an exchange of capabilities has just been made based on one of the criteria listed in section 2.6.3.1, as described in section 2.6.4)
- If an error takes place, after the call when the service was abnormally interrupted or irregular signalling behaviour during the establishment of the call is detected.
- In any of the scenarios described in section 2.6.3.1 which are relevant to the service.

2.6.3.9.2 RCS IP Video Call

In addition to the general user driven entry points and taking into account:

- The optimisations provided in section 2.6.1.5, and,
- The potential impact of having an SPI relationship between sender and receiver as described in section 2.6.2.4.

When enabled, the capability exchange shall take place:

- Before the RCS IP Video Call is initiated by the sender to verify if the receiver is ready for the service (unless an exchange of capabilities has just been made based on one of the criteria listed in section 2.6.3.1, as described in section 2.6.4)
- If an error takes place, after the call when the service was abnormally interrupted or irregular signalling behaviour during the establishment of the call is detected.
- In any of the scenarios described in section 2.6.3.1 which are relevant to the service.

2.6.3.10 Geolocation PUSH

In addition to the general user driven entry points and taking into account:

- The optimisations provided in section 2.6.1.5, and,
- The potential impact of having an SPI relationship between sender and receiver as described in section 2.6.2.4.

When enabled, the capability exchange shall take place:

- Before the initial SIP INVITE is sent to initiate the service to verify if the receiver is ready for the service (unless an exchange of capabilities has just been made based on one of the criteria listed in section 2.6.3.1, as described in section 2.6.4)
- If an error takes place:
    - After the service is cancelled either by the sender or receiver.

- If an error takes place and as a result the Geolocation PUSH is abnormally interrupted or irregular signalling behaviour during the establishment of the service is detected.

2.6.3.11 Geolocation PULL

In addition to the general user driven entry points and taking into account:

- The optimisations provided in section 2.6.1.5, and,
- The potential impact of having an SPI relationship between sender and receiver as described in section 2.6.2.4.

When enabled, the capability exchange shall take place:

- If an error takes place:
  - After the service is cancelled either by the sender or receiver.
  - If an error takes place and as a result the Geolocation PULL is abnormally interrupted or irregular signalling behaviour during the establishment of the service is detected.

2.6.4 Capability exchange optimisations

Depending on the circumstances and use cases, there could be occasions where the capability exchange may happen relatively often (in case of very frequent bearer changes for instance).

To avoid the overhead and increase the efficiency, the client may implement a mechanism to reduce the number of requests in situations where the capability exchange is likely to be performed too often. Examples of how this mechanism can be achieved are listed below:

- Introduce a degree of hysteresis (that is a capabilities update is sent/requested only when the circumstances which led to the change remain stable for a certain period of time).
- Implement a validity timer (that is if the latest capabilities we have were fetched less than X seconds ago, they are still considered as valid).

Please note that this specification does not describe detailed implementations to leave room for OEMs and third parties to drive innovative and differentiated solutions. This helps to distinguish their products from competitors.

2.6.4.1 Service Provider Controlled Service Capabilities Handling

The following items can be configured subject to the Service Provider’s policies (see section A.1.10):

1. The maximum amount of capability query operations during a certain time period done by a client (that is, for all contacts).
2. An expiry of the capabilities for a specific contact.
3. The Contacts considered for the capability discovery depending on their prefix (see Capability Discovery Allowed Prefixes defined in Table 82 in section A.1.10).

This will allow to control the maximum time before a client will discover that one of the contacts is now RCS capable

NOTE: There might be a conflict between the different provisioning settings controlling the frequency of capability query operations (for example, a too low maximum amount of fetch operation combined with a very low expiry time). In that case, an RCS client will prioritise the maximum amount of fetch operations settings over the expiry. A Service Provider deploying RCS is likely to carefully consider the values of these settings and this is therefore not expected to be an issue in actual deployments.

2.7 Capability values and status

The RCS capabilities represent the list of services that an RCS user/client can access at a certain point in time. The capabilities depend on four factors:

1. User Service Provider provisioning status: A Service Provider may choose to limit service to customers depending on subscription status (e.g. chat and file share, but not video).
2. The terminal hardware (HW): A terminal with limited HW (i.e. no capability to process video) may not be able to access all the RCS Services.
3. The terminal status: Even if a terminal HW supports all the services, it could be that the device status introduces a limitation (e.g. receiving files is not possible when the file storage is full).
4. Connectivity status: Some services may require a certain level of network Quality of Service (QoS). For example, streaming video over a 2G GPRS does not provide an adequate UX.

In addition to the factors presented above and as presented in Annex A section A.1, it is possible for a Service Provider to select which services are available for a particular user. Therefore, the previous considerations shall only be taken into account assuming that the relevant RCS services are enabled via configuration and consequently, Table 23 assumes that all the user's devices have been configured with all the RCS services enabled.

As a summary, please find the table below (note that it is assumed the client/terminal and the network supports each of the services as a precondition and that the client/terminal is provisioned to support all the services[13]):

Service	TERMINAL and STATUS REQUIREMENTS	Data Bearer					
		2G	EDGE	3G	HSPA	LTE	Wi-Fi
Standalone messaging	None	Y	Y	Y	Y	Y	Y
Chat (1-to-1 or group)	None	Y	Y	Y	Y	Y	Y

[13] As presented in Annex A section A.1, it is possible for a Service Provider to select which services are available for a particular user.

Service	TERMINAL and STATUS REQUIREMENTS	Data Bearer					
		2G	EDGE	3G	HSPA	LTE	Wi-Fi
File Transfer via MSRP[14]	Minimum threshold of free space to store files	Y	Y	Y	Y	Y	Y
File Transfer via HTTP	The relevant configuration parameters are correctly set	Y	Y	Y	Y	Y	Y
Content share: Image Share	Minimum threshold of free space to store files. The terminal should be on an active call[15] with the user the image is willing to be shared with. Not available in multiparty calls.	Y[16]	Y[16]	Y	Y	Y	Y
Content share: Video Share (IR.74)	Support video profile (encoding /decoding). The terminal should be on an active call[15] with the user the video is willing to be shared with. It is not available in multiparty calls.	N	N	Y One Way Only[17]	Y[18]	Y[18] Higher video profile	Y[18]
SPI	N/A	Y	Y	Y	Y	Y	Y
IP Voice Call [PRD-IR.92]/[PRD-IR.51]	N/A	N	N	N	N	Y (IR.92)	Y (IR.51)

[14] When the Chat Messaging Technology is set to OMA CPM using the CHAT MESSAGING TECHNOLOGY configuration parameter defined in section A.1.4.3, the client shall indicate both the File Transfer and File Transfer Store and Forward capability as described in section 2.6.1.3.1

[15] In this context, the term active call is used to indicate that a voice call is taking place with the user the content is shared with and that this call is not on-hold, waiting or forwarded/diverted. This limitation is not applicable for broadband access devices for the handling of a received capability request or an incoming invitation. The restrictions fully apply for outgoing requests.

[16] Note that it is only possible if device and the cellular network support Dual-Transfer Mode (DTM)

[17] If on the current bearer sharing is supported one way only and a Video Share session is initiated by the device, a capability exchange should be performed to the other end to indicate that Video Share is no longer available. When the session is terminated or the bearer changes to one supporting bidirectional Video Share, the Video Share capability should again be announced.

[18] In this case both ends may share video simultaneously meaning that there is a possibility to have a bidirectional flow of video (see the other party's video while I am also sharing video with him/her). The meaning is that if a user is already sharing video with the other end, the other user may decide to also share video simultaneously, not that the two-ways Video Share can start simultaneously.

Service	TERMINAL and STATUS REQUIREMENTS	Data Bearer					
		2G	EDGE	3G	HSPA	LTE	Wi-Fi
IP Video Call [PRD-IR.94]	Support video profile (encoding /decoding).	N	N	N	N	Y (IR.94)	Y (IR.51/IR.94)
RCS IP Voice Call	N/A	N	N	Y[19]	Y[19]	Y[19]	Y[19]
RCS IP Video Call	Support video profile (encoding /decoding).	N	N	Y[19]	Y[19]	Y[19]	Y[19]
Geolocation PUSH	Minimum threshold of free space to store files From the capability exchange point of view there are no additional terminal requirements however on the sender the service shall be only available if the terminal (UE) provides a mean to access the location information required for the service.	Y	Y	Y	Y	Y	Y
Geolocation PULL	Primary device with capability for locating	Y	Y	Y	Y	Y	Y
Call Composer	The relevant configuration parameters are correctly set The terminal (UE) provides a mean to access the location information required for the service. The terminal (UE) provides a mean to insert subject. The terminal (UE) provides a mean to insert importance.	Y[16]	Y[16]	Y	Y	Y	Y
Post-Call	Support audio message profile (encoding /decoding).	Y	Y	Y	Y	Y	Y
Shared Map	The terminal should be on an active call[15] with the user the map is willing to be shared with. It is not available in multiparty calls.	Y[16]	Y[16]	Y	Y	Y	Y
Shared Sketch	The terminal should be on an active call[15] with the user the canvas is willing to be shared with. It is not available in multiparty calls.	Y[16]	Y[16]	Y	Y	Y	Y

Table 23: RCS services: Terminal, status and data bearer requirements

[19] Only for devices not enabled for VoLTE or VoWiFi and depending on Service Provider Policy

2.7.1 Additional considerations for specific RCS services

2.7.1.1 Chat store and forward: Impact in the capability exchange

As presented in section A.1.4.3 (IM CAP ALWAYS ON), there is the possibility to configure the client to assume that the Service Provider will be providing the Chat store and forward functionality, which consists of storing messages which are sent to users who are offline (i.e. no data connectivity or device off) at the time the chat message is sent.

If this parameter is enabled, there is an impact from the Chat capability which is presented to the user.

As a consequence, we have four different types of contacts for Chat capability:

ID	Targeted contact is RCS Chat capable?	Originating Service Provider supports Store& Forward?	Targeted contact is connected to the network?	Impact on starting Chat
1	NO	N/A	N/A	Chat with that contact is only possible if interworking is provided (see IM CAP NON RCS in Table 75)
2	YES	NO	NO	Not possible to start a Chat at that time
3	YES	YES	NO	Possible to send a Chat message that will be delivered later by the Store and Forward server as soon as the Contact is connected
4	YES	Not Relevant	YES	Chat is possible and messages are immediately delivered

Table 24: Store and forward possible scenarios

The Chat behaviour on the client is controlled by the IM CAP ALWAYS ON configuration parameter (see Annex A for further information): When a Service Provider implements store and forward, they may choose to provision all the RCS users with the IM CAP ALWAYS ON configuration parameter set to enabled. This means that all RCS contacts (currently registered or not) are presented with the Chat service as available (3 and 4 according to Table 24). This may also be the case when the device is offline itself. In that case, the composed messages should be queued and sent as soon as the device registers with the service again. When the messages are included in the SIP INVITE request (see section 3.3.4) a provisional response (including 100 Trying) should be received on the SIP INVITE request before sending a subsequent queued message. If a session is established (i.e. a 200 OK response is received on the SIP INVITE request that may or may not have included a message), the remaining queued messages targeted at that contact shall be sent in the established MSRP session where a following queued message should be sent as soon as a MSRP 200 OK has been received on the last one.

When a Service Provider prefers that the RCS user does not use Chat when an RCS contact is not currently registered in IMS, it sets the IM CAP ALWAYS ON configuration parameter to disabled.

When store and forward is not implemented by the Service Provider, all its RCS customers will have the IM CAP ALWAYS ON configuration parameter set to disabled (2 and 4 according to Table 24).

When interworking of chat to SMS/MMS is available for users, the IM CAP NON RCS parameter can be used in a similar way. More information can be found in section A.1.4.3.

2.7.1.2 Video and Image Share additional considerations

2.7.1.2.1 Bi-directional Video Share

Bi-directional Video Share means that once User A is sharing a video with User B and providing the right coverage conditions are in place, User B could also start to share a video with User A simultaneously. In this case, each Video Share session is independent and is handled separately. When a device moves from a bearer that supports this bi-directional Video Share to a bearer that only supports one way sharing (e.g. from HSPA to 3G) and there is an active Video Share session in each direction, the device that changed bearers shall terminate the Video Share session that it initiated itself.

For clarification purposes, the following assumptions are made for the Image and Video Share cases:

- Both the sharing and receiving end are in a call (that may for instance be CS) between them
- The call is not a multiparty call
- The call is not on hold
- The call is not waiting
- A call forward or divert is not in place

Meaning the relevant Image and Video Share tags described in section 2.6.1.1.2 shall be included only if:

1. The OPTIONS exchange happens when the user is on an active call, and,
2. The destination (sending OPTIONS) or the requester (receiving an OPTIONS message which has to be replied with a response) is on the other end of the active call, and,
3. Network coverage supports sharing (see section 2.7), and,
4. Either bi-directional sharing is supported or the device has not initiated a sharing session itself.

Also for clarification, provided other RCS services (e.g., Standalone Messaging, Chat, File Transfer) are available (e.g. the conditions of coverage and space are met and the device UI supports these services simultaneously with the call), the relevant service capability tags should be included with the Image and Video Share tags.

NOTE: While capability exchange is reciprocal, User A and User B's capabilities may be different and services shall be made available accordingly (e.g. User A may support video encode and User B may support decode, but both need to be under 3G or better data coverage for the service to operate).

In addition to the information presented above, it should also be taken into account that some terminals do not support 2G DTM (Dual-Transfer Mode). When such devices are within a 2G data coverage (meaning that no services are available during the call), the PS connection will automatically drop once they engage in a CS call.

NOTE: Information on codec support for Video Share is covered in section 3.6.

2.8 RCS protocols

The following table summarises the list of protocols employed by RCS clients. It must be noted that the choice among the options presented will not impact Service Provider interoperability:

Protocol name	Description	Transport layer	Secure transport layer/protocol
Session initiation protocol (SIP)	Client-IMS core signalling protocol	User Datagram Protocol (UDP) over IP or Transmission Control Protocol (TCP) over IP	SIP over Transport Layer Security (TLS) or IP Security (IPsec)
Message Session Relay Protocol (MSRP)	Chat messages, media (pictures) and file exchange protocol	TCP/IP	MSRP over TLS
Real-time protocol (RTP)	Real Time Media (voice and video) exchange	UDP/IP	Secure RTP (SRTP) (see [RFC3711])
Internet Mail Access Protocol (IMAP)	Access to Message Store Server	TCP/IP	IMAP over TLS
Hyper Text Transfer Protocol (HTTP)	XML configuration access protocol (XCAP) transactions HTTP configuration mechanism	TCP/IP	HTTPS

Table 25: RCS protocols

According to [RFC3261] RCS clients shall support both SIP/UDP (User Datagram Protocol) and SIP/TCP (Transmission Control Protocol). The choice of whether both are used or only TCP is used to transport the signalling data belongs to each Service Provider and is controlled by the configuration parameters "*psSignalling*", "*psSignallingRoaming*" and "*wifiSignalling*" in [PRD-RCC.15] section 2.2.2.2.

NOTE: The "psSignallingRoaming" parameter is defined as a temporary workaround to address PS roaming related issues identified in live deployments.

Regarding the impact of Network Address Translation (NAT) traversal in the different protocols involved in RCS, the following considerations shall be taken into account:

- Regarding the SIP protocol:
    - Carriage Return Line Feed (CRLF) keep-alive [RFC6223] support is MANDATORY when only SIP/TCP or SIP/TLS is used by the RCS client and SIP/UDP is not used. Section C.1 describes how both client and server could initiate the sending of the keep alives.
    - Simple Traversal of UDP through NATs (STUN) keep-alive [RFC6223] support is RECOMMENDED when SIP/UDP is used by the RCS client as it allows network capacity optimization.
    - An RCS client using SIP/UDP:
        - Shall support symmetric signalling (That is the IP and port combination used to send SIP messages is the same as the one used to receive SIP messages).
        - Shall perform TCP switchover for large SIP messages.
- For handling Message Session Relay Protocol (MSRP) sessions, the RCS MSRP endpoints shall support:
    - [RFC6135]: "The Alternative Connection Model for the Message Session Relay Protocol (MSRP)"
    - The mechanisms described in section 2.8.2 regarding session matching for MSRP.
    - For NAT traversal for MSRP, keep alives (i.e. empty MSRP packets) are not necessary. If the TCP connection is torn down because of inactivity, the MSRP session is torn down, and a new SIP INVITE request to set up a new MSRP session is sent the next time a message is to be sent.
- Regarding NAT traversal of Real-Time Transport Protocol (RTP) sessions, the RCS client should implement the mechanism described in section 2.8.1.
- For Internet Mail Access Protocol (IMAP) and HTTP no specific mechanisms are mandated in the current specification to support NAT traversal.

The support of Transport Layer Security (TLS) based or IP Security (IPsec) based protocols to secure the signalling and TLS based for MSRP and IMAP protocols or Secure Real-Time Transport Protocol (SRTP) for RTP protocols to secure media exchanges is recommended particularly for those scenarios where the data is carried over a network outside the Service Provider domain (i.e. Wi-Fi access). For more information on access security, see section 2.13.

NOTE: To ensure interoperability of all RCS capable devices across different Service Provider networks, the list of preferred options for the transport and security for the signalling and media protocols is included in the configuration parameters as defined in [PRD-RCC.15], section 2.2.2.2. Consequently, a Service Provider provides this information as part of the first-time or re-configuration scenarios described in section 2.3.

2.8.1 RTP and NAT traversal

As mentioned previously, an RCS client must implement several mechanisms to avoid the negative impact of NAT traversal, which can both occur when connecting over:

- PS: Mainly due to the scarcity of IPv4 public addresses and proxying performed at APN level, or,
- Wi-Fi: In this case due to the fact the network topology between the access point and the Internet may vary between deployments.

To combat the negative effects of NAT traversal on the RTP protocol, the RCS client:

- Shall support a keep-alive mechanism to open and maintain the NAT binding alive regardless of whether the media stream is currently inactive, send-only, receive-only or send-receive. The recommended standard keep-alive mechanism is an empty (no payload) RTP packet with a payload type of 20 (as per [3GPP TS 24.229]).

  - SHALL when sending empty packets instead of using STUN and it is about to receive a Video Stream send these dummy RTP packets at a high rate (recommended rate: 50 to 100ms) from the moment the SIP INVITE request is received (or the 180 RINGING is sent) in bursts sent regularly (a 1 second burst every 15 seconds is recommended). This shall be done until one of the following conditions is met:

    - The first RTP packet of a Video Stream is received, or,
    - The client starts streaming itself in case of a bi-directional RTP stream, or,
    - A final response is sent on the SIP INVITE request. In case this final response is a 200 OK response, the client shall continuously send the dummy RTP packets until either the first RTP packet of a Video Stream is received or the client starts streaming itself in case of a bi-directional RTP stream.

    Once the first RTP packet is received the dummy packets shall be sent at a lower rate (a transmission every 15 sec is recommended) for the remainder of a uni-directional session or not at all in case the RTP stream is bi-directional.
  - If the first frame is not an I-Frame or Network Abstraction Layer (NAL) unit carrying a Sequence Parameter Set (SPS) or Picture Parameter Set (PPS), the receiving client SHALL send a RTCP Full Intra Request (FIR) (see [RFC5104], section 4.3.1) to the sender.
  - SHALL reset the encoder as specified in [RFC5104] when receiving an RTCP FIR, and send SPS, PPS (if not provided in the SDP) and an I-Frame to the receiver.

- Shall use symmetric media (that is use the same port number for sending and receiving packets) as defined in [RFC4961] mechanism which is summarized below:

  - When an invitation for Video Share is received and accepted, the 200 OK response contains a SDP body containing all the necessary fields (including the destination port) for the sender to send the RTP packets.
  - Immediately after sending the 180 Ringing response, the receiver will send a keep-alive packet back to the sender to secure the timely setup of the media path:
  - The source port shall be identical to the one included in the m field of the SDP payload inside the 200 OK response.
  - The destination port shall be identical to the one included in the m field of the SDP payload inside the SIP INVITE message.

    - The sender should allow enough time for the media path to be secured.

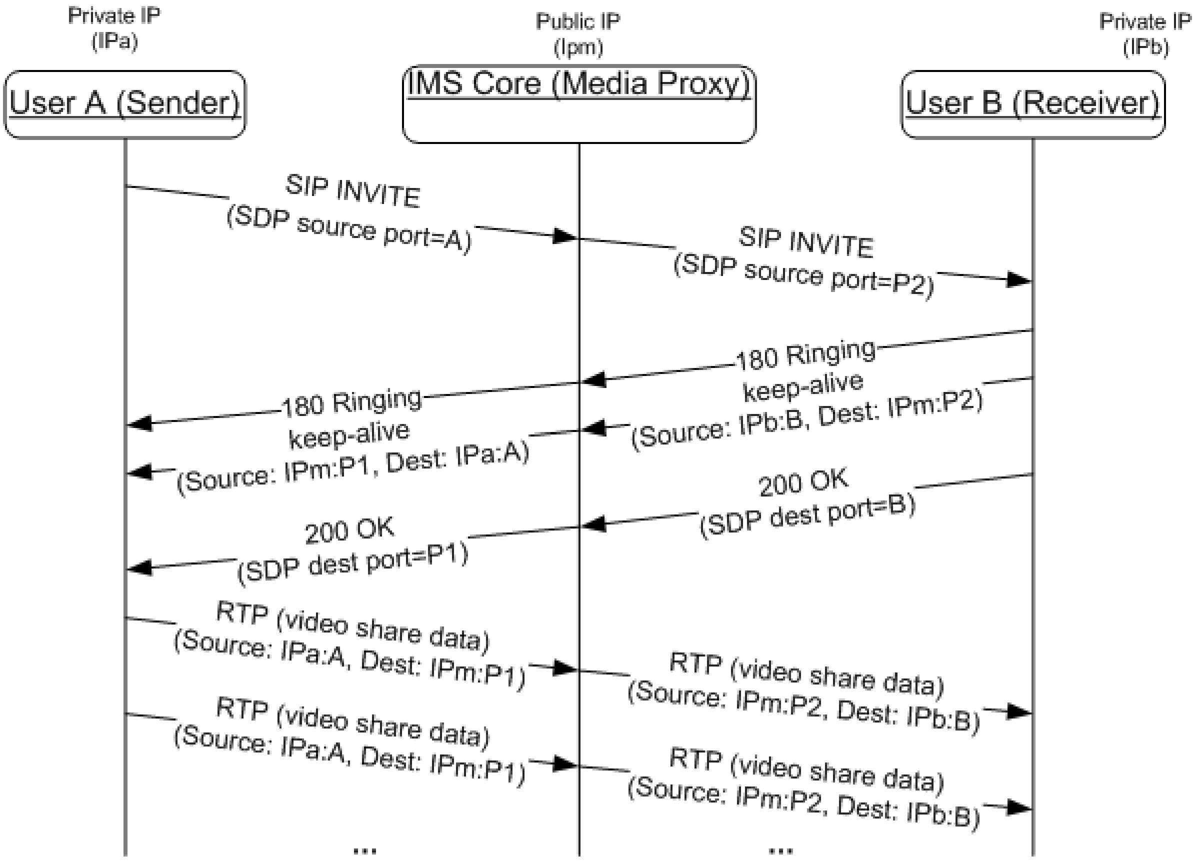

Figure 17: RTP symmetric media path establishment

NOTE1: as a general recommendation, User A should also send a keep-alive once it receives the SDP from the other side.

- Shall use the Real-Time Transport Control Protocol (RTCP):

  The symmetric media procedure described for the RTP protocol is, in general, applicable to any UDP stream. As the usage of RTCP is also mandatory, an analogous mechanism shall be implemented to prevent any RTCP streams from being blocked. Therefore, the symmetric media procedure described in this section for RTP is also applicable to RTCP and shall be employed (that is a dummy packet is sent by the receiver to secure the RTP flow and a second one is used to secure the RTCP flow). Also the sender device/client shall send a dummy packet when the session is established to secure the RTCP flow on their side and ensure the reception of any RTCP RR (Receiver Report) sent by the receiving side. The dummy packet format recommended for establishing the RTCP flow is an empty RTCP RR or empty RTCP SR (Sender Report).

NOTE2: For a VoLTE/VoWiFi enabled device, RTCP usage for a voice session shall be as defined in section 3.2.4 of [PRD-IR.92]

Please note that for readability purposes, the procedures described in this section have not been included in the diagrams in section 3.6 covering the Video Share functionality.

2.8.2 MSRP session matching

NOTE: The text in this section is based on the text contained in the now expired IETF internet draft draft-ietf-simple-msrp-sessmatch-10.

When the CHAT MESSAGING TECHNOLOGY configuration parameter defined in Table 75 is set to use OMA CPM, an RCS client shall set up MSRP sessions as per [RCS-CPM-CONVFUNC-ENDORS]. Otherwise, the session shall be set up according to the procedures in [RFC4975], [RFC6135] and the procedure in this section for the session matching.

This section defines how an MSRP entity (e.g. an RCS Client, Messaging Server or other node handling MSRP within the network) that does not support the procedures in [RCS-CPM-CONVFUNC-ENDORS] or is configured not to use them matches an incoming MSRP message to an MSRP session. The difference between the session matching mechanism in [RFC4975] and the one defined here is that while the mechanism in [RFC4975] uses the MSRP URI comparison rules for session matching, for RCS, only the session-id part of the MSRP URI is used.

When an MSRP entity receives the first MSRP request for an MSRP session, the To-Path header field of the request should contain a URI with a session-id part that was provided in the SDP associated with the MSRP session. The entity that accepted the connection looks up the session-id part of the MSRP URI in the received requests, in order to determine which session it matches. The session-id part is compared as case sensitive. If a match exists, the entity shall assume that the host that formed the connection is the host to which this URI was given. If no match exists, the entity shall reject the request with a 481 error response. The entity shall also check to make sure the session is not already in use on another connection. If the session is already in use, it shall reject the request with a 506 error response.

2.8.3 SIP Issues

1. An RCS client should use a random originating SIP signalling port of the range 1025-65535. If the selected port is not available, the next port number shall be used for this session.

2. An RCS client shall build its SIP contact address to be unique. A recommended way to do so is to use a hashed value of the +sip.instance tag as user part of the URI of the contact address.
3. For an incoming request, an RCS client should verify that the Request-URI matches the URI of its registered contact address. If not, the Request-URI shall be considered an unexpected address and the request shall be rejected as per [RFC3261] section 8.2.2.1.

2.9 RCS and Access Technologies

2.9.1 RCS and Cellular/EPC-integrated Wi-Fi Access

2.9.1.1 Access used by RCS in relation to VoLTE/VoWiFi

A device providing both RCS and VoLTE/ViLTE or VoWiFi shall support the procedures described in [PRD-NG.102]. These procedures are dependent on the RCS VOLTE SINGLE REGISTRATION configuration parameter defined in section A.1.7.

2.9.1.2 LTE Radio Capabilities

Radio bearers, UE Discontinuous Reception (DRX) and Discontinuous Transmission (DTX) modes of operation, Radio Link Control (RLC) configurations, and Guaranteed Bitrate (GBR) and Non-Guaranteed Bitrate (NBGR) services and GBR Monitoring Function are all as specified in [PRD-IR.92] for devices enabled for VoLTE. None of this is applicable to other devices and access networks other than LTE.

2.9.1.3 Bearer aspects

For all IMS traffic the following applies for an RCS device configured for VoLTE/VoWiFi:

- For LTE bearer management see section 4.3 of [PRD-IR.92] respectively.

For all RCS IMS traffic the following applies:

- For a device enabled for VoLTE: LTE QCI (QoS class identifier) 8 and 9 shall be supported so that either may be used for MSRP traffic.
- For a device using the IMS APN for RCS (see section 2.9.1.4): LTE QCI (QoS class identifier) 8 and 9 shall be supported so that either may be used for MSRP traffic.
- For other devices: no requirements.

2.9.1.4 APN and roaming considerations

General technical guidelines on how roaming is handled for the RCS services shall follow [PRD-IR.65].

Guidance given for RCS and access technologies as documented in section 2.9 are applicable also in the roaming scenarios. Specific roaming considerations for the different RCS device types (as specified in section 2.2):

- All services on a primary device enabled for VoLTE, shall follow the general rules as per [PRD-IR.88], APN usage as per [PRD-NG.102].
- All services on a primary device enabled for VoWiFi shall follow the general rules as per [PRD-IR.61], APN usage as per [PRD-NG.102].
- Other devices: no specific requirements.

The APN to be used to access the RCS services[20] depends on the capacity of the device and the network to support an IMS APN as per [PRD-IR.88], on the device configuration and on the client type (see section 2.2):

- When the device is configured for VoLTE or VoWiFi, an embedded client shall use the APN indicated in [PRD-NG.102].
- For an embedded client on other devices, the behaviour shall depend on the setting of the RCS VOLTE SINGLE REGISTRATION (see section A.1.6) and NO MSRP SUPPORT configuration parameters (see section A.1.12):

NOTE: The RCS VOLTE SINGLE REGISTRATION configuration parameter is used in this case even if there is no VoLTE registration from the device because the required behaviour is similar.

  - The IMS APN shall be used to access the RCS services when the device is configured through the RCS VOLTE SINGLE REGISTRATION configuration parameter defined in section A.1.6 to use a single registration approach.
  - The Home Operator Services (HOS) APN shall be used to access the RCS services when the device is configured through the RCS VOLTE SINGLE REGISTRATION configuration parameter defined in section A.1.6 to use a dual registration approach.
  - When roaming on a network that is listed in the NO MSRP SUPPORT configuration parameter (see section A.1.11), the client shall ignore the value of the RCS VOLTE SINGLE REGISTRATION configuration parameter when selecting the APN to use. The client shall use the HOS APN if obtained through client configuration; otherwise, the client shall use the Internet APN[21].

NOTE1: In the longer term, the visited network should indicate to the UE whether MSRP support is available. Until this is standardized, the NO MSRP SUPPORT configuration parameter is used.

NOTE2: When roaming on a network where the device cannot access a local IMS APN (e.g. no VoLTE roaming agreement is in place), a client configured to use the IMS APN will, by using the IMS APN, automatically access RCS through the home network's IMS APN with the telephony service using the Circuit Switched network. If no IMS roaming agreement is in place, the visited network would degrade any requested QOS based on local configuration, as per normal procedures in the MME for any APN as specified in [PRD-IR.88]. The operator could optionally have a QoS data roaming agreement to ensure that QCI=5 for the IMS APN is allowed in the visited network and that either QCI=8 or QCI=9 for MSRP will be allowed on the IMS APN. If the appropriate QCI bearer cannot be set up for MSRP

[20] This section only covers the APN behaviour for RCS services. These settings shall not be taken into account for the usage of other APNs by non-RCS services.

[21] By internet APN, we understand the default APN configured by the Service Provider to provide Internet connectivity on the device

when on LTE access, MSRP traffic will be on the default QCI=5 bearer unless prevented by the terminal or network.

To support traffic from non-RCS applications (e.g. generic internet access) in this case the device and network shall support other APNs to be active simultaneously.

For an embedded client, the APN to use for HTTP, XCAP and IMAP shall be the HOS APN as defined in [PRD-IR.88]. The HOS APN can be configured to be or by default be, the device's generic data access APN (i.e. the internet APN[21]). The network should use the HOS APN when providing the RCS client with its configuration, in order to prevent unwanted data charging for client provisioning traffic.

It is out of scope of this specification how the HOS APN is configured on the device.

- Downloadable clients shall use the internet APN.

2.9.1.5 Data Off

Users can switch cellular data usage off locally on their device. To allow the operator to offer IR 92 / IR 94 and RCS services to their customers even in these use cases, the data off switch shall have an operator configurable impact on the device connectivity. The service provider should ensure a good service experience if IP service usage is allowed although the data switch was set to off by the end user. The parameters to do so are introduced in section A.1.15 and are applicable for primary devices when RCS is not using the internet APN (see section 2.9.1.4).

When the cellular data switch is switched off and as a consequence a service is disabled according to the configuration in section A.1.15, the client shall not include the corresponding tags and service identifiers in the registration and capability exchange.

When according to the configuration defined in section A.1.15 all RCS services being relevant for capability discovery are disabled when the cellular data switch is switched off (i.e. all of RCS MESSAGING DATA OFF, FILE TRANSFER DATA OFF, CONTENT SHARE DATA OFF, PRE AND POST CALL DATA OFF and EXTENSIONS DATA OFF are set to 0 or are set to 2 and the device is connected to a cellular network other than the HPLMN or the parameters are not relevant because the corresponding service is not available on cellular networks), a client shall also disable the capability exchange when needing to use cellular networks to provide those services and cellular data is switched off. In those circumstances, a client shall neither send capability exchange requests nor respond to such requests (allowing them to time-out in the network).

When according to the configuration defined in section A.1.15, all IMS services are disabled when the cellular data switch is switched off (i.e. all of RCS MESSAGING DATA OFF, FILE TRANSFER DATA OFF, SMSOIP DATA OFF, CONTENT SHARE DATA OFF, PRE AND POST CALL DATA OFF, VOLTE DATA OFF, IP VIDEO CALL DATA OFF and EXTENSIONS DATA OFF are set to 0 or are set to 2 and the device is connected to a cellular network other than the HPLMN) or are not relevant because the corresponding service is not available on cellular networks and provided the client does not offer any other IMS services beyond the scope of VoLTE and RCS, a client shall not register in the IMS when needing to use cellular networks to provide those services and cellular data is switched off. In this case, if the device is in LTE coverage, the device shall either connect to 2G/3G network access only or in LTE keep a data connection over which no data is sent if no other data connection is required/available for the Circuit Switched Fall Back and SMS over SGs to happen.

For services using non IMS protocols having no data connection at the time of disabling, no additional actions are required by the client. If disabled by Data off configuration the service will not be available at the time of invocation.

For services using non IMS protocols and having a data connection active at the time of disabling, the ongoing session or transaction shall be terminated.

2.9.1.6 Summary of conditions and parameters that control the access network used

The combination of the switches, configuration parameters and coverage conditions introduced to control the connection through which the service is delivered leads to the behaviour described in Table 26:

#	Telephony coverage	Wi-Fi Coverage	Registration approach following from RCS VOLTE SINGLE REGISTRATION[22] and NO MSRP SUPPORT[23]	RCS Service in Data Off[24]	Cellular Data Switch	Used network for RCS[25]
1	VoLTE	N/A	Single Registration	N/A	On	Cellular (IMS APN)
2	VoLTE	N/A	Single Registration	On	Off	Cellular[26](IMS APN)
3	VoLTE	N/A	Single Registration	Off	Off	None (RCS unavailable)[26]

[22] See **Error! Reference source not found.**

[23] Dual registration is used when the RCS VoLTE SINGLE REGISTRATION configuration parameter is configured to 0, when RCS VoLTE SINGLE REGISTRATION configuration parameter is configured to 2 and the device is roaming and when roaming on a visited network that is listed in the NO MSRP SUPPORT configuration parameter. Otherwise single registration is used.

[24] i.e. at least one of RCS MESSAGING DATA OFF, FILE TRANSFER DATA OFF, CONTENT SHARE DATA OFF, PRE AND POST CALL DATA OFF, EXTENSIONS DATA OFF as defined in A.1.15 is set to 1 or it is set to 2 and the device is attached to the HPLMN or IP VIDEO CALL DATA OFF defined in A.1.15 is set to 1 or it is set to 2 and the device is attached to the HPLMN on a device that is not enabled for VoLTE.

[25] i.e. for traffic related to Standalone Messaging, 1-to-1 Chat, Group Chat, File Transfer, Video Share and Image Share, RCS IP Voice Call, RCS IP Video Call and Extension to Extension traffic as defined in section 3.2, 3.3, 3.4, 3.5, 3.6, 3.8, 3.9 and 3.12 respectively. VoLTE, ViLTE, SMSover IP and MMS always use the cellular network.

[26] Case assuming VoLTE remains switched on when data is off. If not, available cellular coverage is assumed to be CS.

#	Telephony coverage	Wi-Fi Coverage	Registration approach following from RCS VOLTE SINGLE REGISTRATION[22] and NO MSRP SUPPORT[23]	RCS Service in Data Off[24]	Cellular Data Switch	Used network for RCS[25]
4	CS	No	Single Registration	N/A	On	Cellular (IMS APN)
5	CS	No	Single Registration	On	Off	Cellular (IMS APN)
6	CS	No	Single Registration	Off	Off	None (RCS unavailable)
7	CS	Yes	Single Registration	N/A	N/A	Non EPC-integrated Wi-Fi
8	VoWiFi	Yes	Single Registration	N/A	N/A	EPC-integrated Wi-Fi
9	VoLTE or CS	No	Dual Registration	N/A	On	Cellular (HOS APN)
10	VoLTE or CS	No	Dual Registration	On	Off	Cellular (HOS APN)
11	VoLTE or CS	No	Dual Registration	Off	Off	None (RCS unavailable)
12	VoLTE, VoWiFi or CS	Yes	Dual Registration	N/A	N/A	Non EPC-integrated Wi-Fi
13	None	No	N/A	N/A	N/A	None (RCS unavailable)
14	None	Yes	N/A	N/A	N/A	Non EPC- integrated Wi-Fi

Table 26: APN configuration proposal for data traffic and roaming

2.9.2 Other access networks

2.9.2.1 Overview

In addition to the cellular PS access networks described in sections 2.9.1.4 and 2.9.1, the RCS framework and services can be used over any IP access over which the Service Provider's IMS core and application servers can be reached, provided that it offers sufficient bandwidth and an acceptable latency. Section 2.7 provides a guideline for which services can be used when connected through different types of access networks including broadband access.

These other networks can be both trusted and untrusted networks. "Trusted" means that HPLMN considers the access network trusted independently of the specific mechanism for authentication and encryption of end-user traffic that the access network implements. The access network is integrated into the Service Provider core infrastructure in such a way that the whole path from a mobile to services is considered secure by the HPLMN and under the control of respective Service Providers. Fixed access networks including ADSL (Asymmetric Digital Subscriber Line), cable modem access, FTTH (Fibre To The Home) and WLAN networks could therefore be considered as a "Trusted" network by the HPLMN if they are entirely Service Provider controlled. The same holds for clients using a cellular PS connection as broadband access.

Untrusted broadband networks, however, are at least partly controlled by some 3rd party and may, therefore, require more elaborate security measures to guarantee privacy and authenticity of the signalling and media traffic. As in this case, the network does not provide the support for functionality such as encryption natively, it needs to be added to it before that particular network can be used to access the IMS core system. If the HPLMN considers direct access from a broadband network such as public Wi-Fi hotspots to the IMS core system untrusted, then in order to avoid security risks such as Denial of Service attacks towards Service Provider core components there is a requirement for additional secure access mechanisms to be deployed.

Many such secure access mechanisms are possible and can either be xSIM based or use other types of credentials. For commercial deployments, the choice will be dependent on the type of client and on the environment. For example, the same type of client could only use PS Mobile Broadband Access, access over the internet or only over a fixed ADSL line / FTTH access which may require using different mechanisms for each case. Therefore given that this choice will end up being specific to each deployment, it is not considered in scope of RCS to specify an exhaustive list of supported access mechanisms.

As described in section 2.8 both trusted and untrusted networks need to be able to provide access and authentication over NAT. This must be taken into account for example when xSIM based access, IPSec or other fixed access authentication mechanisms are used.

This support for access over non-cellular networks can be used in two ways:

1. As an offloading capability for the cellular network
   This will be controlled by the device itself:

   - When a device is enabled for VoLTE/VoWiFi (see section 2.9.1.1), it is expected that the device either remains on LTE access for as long as it is available or switches to EPC-integrated Wi-Fi (see Table 26).
   - When the voice service is provided via CS access, it is up to the device when and whether to move to non-cellular (e.g. non EPC-integrated Wi-Fi) access. If a device moves to a non-cellular network, it is expected that the device first de-registers in IMS from the cellular network, and then registers in IMS in the non-cellular network, or vice versa when moving in the other direction.

2. As a means of access for dedicated broadband clients using the identity of the mobile device
   This can be either as a standalone client when there is no mobile device using that same identity or as secondary client to a mobile device sharing the same identity (see chapter 2.5). In the latter case the user will have multiple devices sharing the same identity. Chapter 2.11 provides further details on how this can be realised. These differences are further detailed in section 2.9.2.2.

2.9.2.2 Dedicated RCS clients on Broadband Access

Next to clients using mobile access, RCS also supports dedicated clients using broadband access. Such a client can operate in two significantly different modes:

1. As a secondary client, adding performance (such as larger keyboard, a screen with higher resolution and so on) to the primary mobile client with RCS functionality. Such a secondary client is designed with user experience aspects, storage accessibility and so on, but is not designed to act as a primary telephony device. In this case the primary client retains aspects a user would associate with their device, for example regulatory functions, quality of service and full access to the telephony functions.

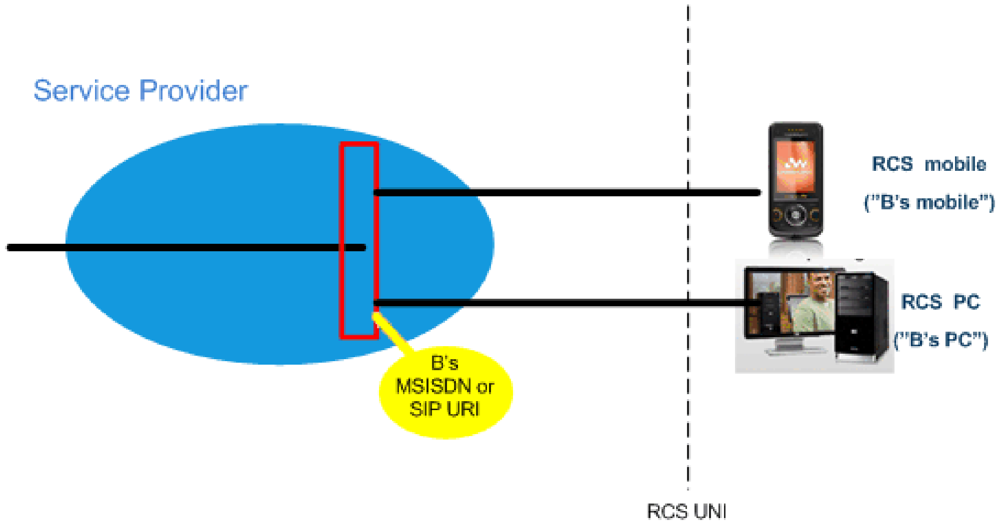

Figure 18: RCS Broadband Access client used as a secondary client

NOTE: Other combinations of multiple devices, such as support of multiple mobile clients, are out of scope for RCS. However, this does not restrict a Service Provider to deploy proprietary solutions to achieve this.

2. As a primary client, replacing the user's mobile client. A primary client has to meet all regulatory requirements (emergency calling, lawful intercept, etc.) and perform to meet the traditionally expected telephony functionality and demonstrate the reliability, performance and quality of service of a primary device. The precondition for its use is that basic telephony services are already available in the Broadband Access network. For these services, the local regulations are already fulfilled.

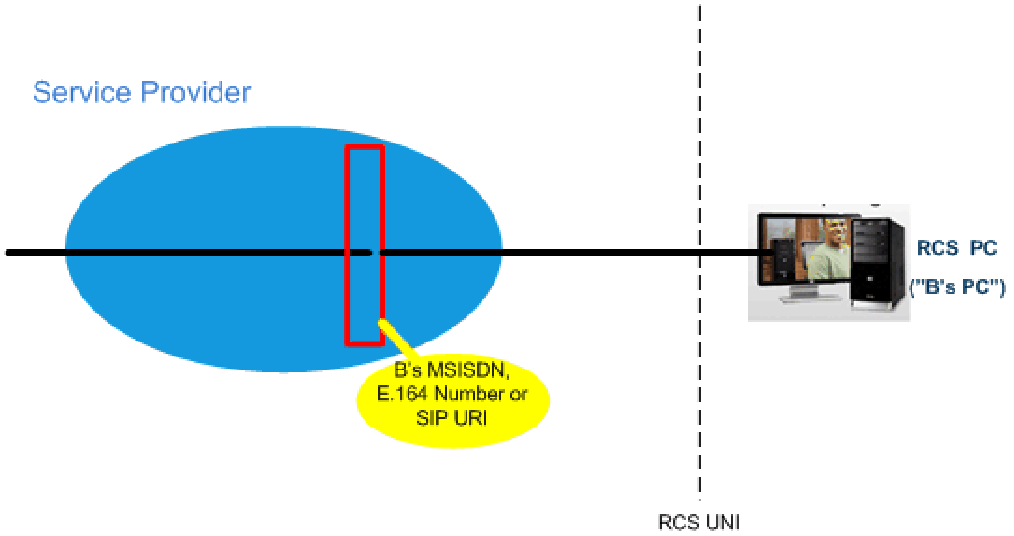

Figure 19: RCS Broadband Access client used as a primary client

2.10 End User Confirmation Requests

RCS clients shall support the End User Confirmation Request enabler as defined in [PRD-RCC.15] section 3.1.

The following list shows the defined system requests in this RCS specification in addition to the generic ones described in [PRD-RCC.15]:

Type	Data	Action
urn:gsma:rcs:extension:control	List of (<IARI>,<duration>) separated by ';'	Prevent each Extension from the list identified by their IARIs to access the RCS infrastructure for the duration provided (in seconds). A duration of 0 indicates that the Extension shall be permanently prevented from using the RCS infrastructure. A <IARI> is formatted in the following way: *urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.ext.<identifier>* example: *(urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.ext.A5TgS99bJloIUIh1209SJ82B21m87S1B87SBqfS871BS8787SBXBA3P45wjp63tk,0);(urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.ext.VoxgS94bJloIUIh12r9Sop1j21m87Spt83SZqfS871BS128pSB2B13P42wjp43rt,3600)*

Table 27: List of System Requests in RCS

2.11 Multidevice support

2.11.1 Overview

As shown in section 2.9.2.2, the use of a broadband access client leads to the possibility of the user having multiple devices that share the same (public) identity, a MSISDN for instance. Depending on the services that are deployed, this multidevice environment allows a user to:

- Answer a call or respond to a message from a device/client that suits their purpose.
- Have a single buddy list shared between the devices/clients.
- Authorise invitations to share Social Presence Information from every device/client.
- Have a single Social Presence Information that can be seen and maintained from every device/client that is used.

The general communication behaviour in this environment is that when the recipient has multiple devices/clients in use and a call or a message is received every recipient's device will alert. The recipient may then respond to the call or to the message from any of their devices; whichever device is the best for the current situation. In addition, when the recipient accepts or rejects a call from any of the devices, all the other devices will stop alerting.

To achieve this, an RCS client shall send a SIP 603 Decline response to the invite request when an RCS User explicitly declines a session invitation for a SIP session based service like for example an IP Voice Call, File Transfer, Video and Image Share. According to [3GPP TS 24.229] and [RFC3261] both such a rejection and an acceptance will result in a

SIP CANCEL request sent by the S-CSCF to the other devices of the user that have not yet accepted nor rejected the invitation. In both cases, the requests may carry a Reason header field as specified in [RFC3326] that is populated with the proper SIP response code values (as per [3GPP TS 24.229]), in this case either the *cause=200* or *cause=603* values.

If the user device has accepted the INVITE with a 200 OK, then the S-CSCF should set the Reason header field with *SIP* protocol and the protocol-cause set to *200* along with an optional protocol-text (e.g. *SIP;cause=200;text="Call completed elsewhere"*).

In case one device has sent a 603 Decline then the S-CSCF should set the *cause=603* along with an optional protocol-text (e.g. *SIP;cause=603;text="Decline"*), in either SIP CANCEL and/or SIP BYE, towards the remaining user devices.

When a client receives a SIP CANCEL request containing a Reason Header field with the protocol set to “SIP” and the protocol-cause set to 200, a client may for example use this information to indicate to the user that the session was accepted on another device (rather than as for example a missed call).

As a fallback for legacy services where this general communication behaviour cannot be realised a call or message might be directed to a certain device.

2.11.2 Addressing of individual clients

Any Application Server (e.g. a Messaging Server or OPTIONS AS) can address an individual RCS client using information received with the third party registration (public GRUU or sip.instance).

If a client obtains GRUUs from the registrar as described in section 2.4, the public GRUU shall be used as device identifier. The client shall use the public GRUU as a URI parameter for the client in non-REGISTER requests and responses that it sends, for example, an INVITE request and 200 OK response where the GRUU will be included in the Contact Header.

If a client does not obtain a GRUU from the registrar, the sip.instance feature tag and value shall be used as the device identifier. The client shall include the sip.instance feature tag in the Contact header with the same instance-id value in any non-REGISTER request and responses that it sends.

2.11.2.1 Backward compatibility

Earlier versions of RCS allowed the RCS client to include its device identifier information in the CPIM From header in a Standalone Message or in an MSRP SEND sent as part of a Group Chat. This is no longer required, so an Application Server may remove this information from the CPIM From header if it is received by an RCS client compliant to an older version of RCS. Of course any IMDNs sent during an ongoing Group Chat will end up arriving at the same device that sent the message.

2.11.3 Routing RCS SIP requests to RCS Clients

In the context of multi-client and multi-device deployments, it is possible that the same IMS public user identity (IMPU) is used by the RCS Client and by other IMS Clients (e.g. Voice over LTE client using SMS over IP), but carrying a different instance ID.

In the messaging case, to ensure that requests are forked only to RCS Clients which have explicitly registered with the required RCS capabilities, the terminating Messaging Server may add a dedicated Accept-Contact header field to each RCS SIP request carrying either a CPM ICSI, or a SIMPLE IM feature tag, with the *require* and *explicit* parameters described in [RFC3841]. This applies to the following services:

- Chat
- Standalone Messaging (text and multimedia messaging)
- File Transfer

2.12 Interconnect principles and guidelines

The Service Provider's IMS NNI shall follow the provisions in [PRD-IR.65] sections 3, 4, 5 and 6.

The Service Provider's RCS NNI shall follow the provisions in [PRD-IR.90]. The implementation could be any of the three connectivity options for RCS NNI defined in [PRD-IR.90].

2.13 Security and privacy

2.13.1 Access Security for the User-to-Network Interface (UNI)

2.13.1.1 Access Signalling Security Methods

Several SIP signalling access security and authentication methods are specified in [3GPP TS 33.203] and [3GPP TS 24.229] for access to the IMS core and IMS based services such as RCS. The applicability and choice of method is highly dependent on the RCS client and access type (e.g. trusted or untrusted) including what is supported or required by the IMS core.

2.13.1.1.1 IMS AKA with IPsec

IMS AKA with IPsec is the preferred long term approach in IMS for access signalling security from a cellular PS network. Such access requires the IMS client device to possess an AKA based credential (e.g. Universal SIM (USIM)/IP Multimedia Services SIM (ISIM)) and support the "*ipsec-3gpp*" procedures specified in [3GPP TS 33.203] and [3GPP TS 24.229].

IMS AKA with IPsec is the access signalling approach specified for Voice over LTE (VoLTE) ([PRD-IR.92]).

2.13.1.1.2 SIP Digest Authentication and TLS

SIP Digest is a username and password challenge based authentication (based on HTTP Digest) which is suited for broadband access to IMS or for RCS clients which do not possess AKA based credentials (e.g. xSIM) or do not support IMS AKA based IPsec. SIP Digest is widely implemented in Internet Engineering Task Force (IETF) based SIP clients and is often deployed with TLS. Support for SIP Digest with and without TLS is specified in [3GPP TS 33.203] and [3GPP TS 24.229] for access to IMS from "non-3gpp" defined access networks (e.g. broadband/fixed access networks).

When an RCS client is enabled for SIP Digest authentication, the client will use the pre-configured SIP username and password as specified in Table 79 to authenticate to the IMS core. For the initial SIP REGISTER message (before a digest challenge), the RCS client shall include an authorisation header (as per [3GPP TS 24.229]) which includes the SIP digest username and an empty digest authentication response parameter. This allows the IMS core to treat the SIP digest username as an IMS private user identity (IMPI) which is distinct from the IMS public user identity (IMPU), allowing the same SIP public user identity (or IMPU) to be registered from multiple RCS clients/devices.

The IMS registration flow for SIP digest authentication is shown in Figure 20. In this example flow, the RCS client is connected to the IMS core over a Wi-Fi internet broadband connection.

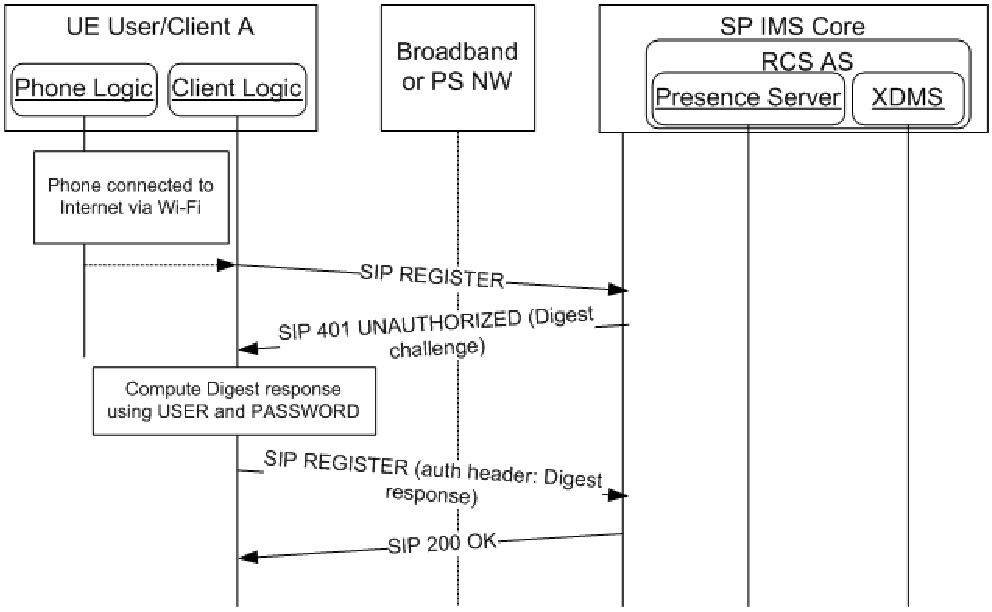

Figure 20: Registration with SIP Digest Authentication

If digest authentication fails two times with a SIP 401 UNAUTHORISED response, the client shall not attempt further registration attempts, but rather consider the current configuration as invalid and force a reconfiguration using the procedures in chapter 2.3 the next time the handset is rebooted.

The use of SIP Digest with TLS is recommended for access from untrusted access networks (including WLAN with no encryption). TLS provides per message authentication, integrity protection and encryption for SIP signalling. TLS with server side certificates also provides authentication of the IMS core to the RCS client.

NOTE: This requires the client to possess a root or intermediate certificate of a Certificate Authority (CA) that is in the certificate signing chain for the IMS core's (e.g. P-CSCF) TLS certificate.

When an RCS client is enabled to use SIP/TLS it should use the SIP TLS port obtained through P-CSCF discovery procedures (e.g. through DNS SRV records [Service records]) or configuration. However, if RCS client is not able to determine a SIP TLS port through these means, it shall use the default SIP port for TLS as specified in [RFC3261].

The RCS client enabled to use SIP/TLS should first use the SIP security agreement (sec-agree) [RFC3329] as specified in [3GPP TS 24.229] to first negotiate the use of TLS with its

SIP Proxy (P-CSCF). Alternatively, an RCS client may first try to establish a TLS session with the SIP proxy (P-CSCF) before sending an initial SIP Register message which does not include sec-agree for TLS. However, with this approach the S-CSCF may challenge subsequent non-Register messages with a 407 Proxy Authentication Required unless configured to trust SIP Digest without signalling security indicated or if the P-CSCF is able to provide this indication despite not using sec-agree.

NOTE: In both cases SIP proxy (P-CSCF) authenticates to the RCS client using a TLS server certificate.

When SIP Digest is not used with TLS, the IMS core may require non-REGISTER SIP requests to be authenticated using the same SIP Digest challenge mechanisms used during registration. However, in this case the SIP digest challenge is sent in a 407 (Proxy Authentication Required) response. An RCS client that receives a 407 (Proxy Authentication Required) response shall respond by sending an authenticated SIP request which includes a Proxy Authorization header with the digest response. The RCS client shall cache the digest challenge data (e.g. server nonce) for use in authenticating subsequent SIP requests using a nonce-count value (for replay protection) as per [RFC2617] and including a Proxy Authorization header with an updated digest response. This avoids the need for the IMS core to challenge each SIP request before the authentication data expires. Once the digest authentication data expires a new challenge will be issued.

NOTE: The IMS core may also support binding the RCS client's IMS identities authenticated during registration with a source IP address (and port if [RFC5626] "SIP Outbound" is used). In such cases, the IMS core may not require subsequent non-registration based SIP messaging to be authenticated using SIP Digest if the identities and source addresses in the messaging matches the binding obtained during the Digest authenticated registration process.

2.13.1.2 Access Signalling Security Profiles for RCS

As there are several considerations which access signalling security method should be used for access to RCS services, the following table defines authentication and access signalling security mechanisms as per RCS device and access type.

Device	Access	Applicable Security Methods	Applicability and Suitability
Mobile client not configured for VoLTE/VoWiFi	Cellular PS Access	SIP Digest (with or without TLS) or IMS AKA with IPsec	IMS AKA with IPSec may be used when supported by both device and the network. SIP Digest with or without TLS is used depending on (pre-)configuration
	access over EPC-integrated Wi-Fi	SIP Digest (with or without TLS) or IMS AKA with IPsec	IMS AKA with IPSec may be used when supported by both device and the network. SIP Digest with or without TLS is used in cases when pre-configured

	Non-cellular broadband (Wi-Fi) access	SIP Digest, SIP Digest with TLS or IMS AKA with IPsec	SIP Digest with TLS is recommended over SIP Digest without TLS SIP Digest with or without TLS is used in cases when pre-configured or when the mobile device does not support IMS AKA for WLAN access
VoLTE/VoWiFi configured mobile client	Cellular PS Access	SIP Digest, SIP Digest with TLS or IMS AKA with IPsec[27] as specified in [PRD-NG.102] NOTE: The configuration to any other method is not possible.	AKA credentials stored securely in a UICC such as an xSIM. The method used depends on the relation to the registration for VoLTE/VoWiFi (see [PRD-NG.102])
	EPC-integrated Wi-Fi	SIP Digest, SIP Digest with TLS or IMS AKA with IPsec[27] as specified in [PRD-NG.102] NOTE: The configuration to any other method is not possible.	AKA credentials stored securely in a UICC such as an xSIM. The method used depends on the relation to the registration for VoLTE/VoWiFi (see [PRD-NG.102])
	Non-cellular broadband (Wi-Fi) access	SIP Digest, SIP Digest with TLS or IMS AKA with IPsec[27].	SIP Digest with TLS is recommended over SIP Digest without TLS SIP Digest with or without TLS is used in cases when pre-configured or when the mobile device does not support IMS AKA for WLAN access.
Broadband Access Enabled		SIP Digest or SIP Digest with TLS	SIP Digest with TLS is recommended over SIP Digest without TLS SIP Digest is used for mobile devices which do not support IMS AKA for WLAN access.

Table 28: Access Signalling Security Profiles for RCS

For RCS devices which can access the IMS core from both mobile and broadband/fixed networks (e.g. Wi-Fi) a separate access signalling security method and corresponding authentication credential may be required. If the security mechanism is not pre-configured

[27] Requires UDP encapsulation of IPsec for NAT traversal

as per section 2.2.1.1.2 and 2.2.2.1.3 of [PRD-RCC.15], the RCS device negotiates the set of security mechanisms using the SIP security agreement [RFC3329] as specified for IMS in [3GPP TS 33.203] and [3GPP TS 24.229]. If the client is pre-configured with a specific access signalling security mechanism, the client uses the signalling corresponding to this security method in the initial registration procedure, and the IMS core determines (based on signalling) which mechanism is being used/requested and then determines (based on security policy) if the access signalling security method is allowed.

NOTE: The RCS device shall support a configuration option for each of these profiles (where applicable).

2.13.1.3 Access Media Security

2.13.1.3.1 Secure RTP (SRTP)

SRTP [RFC3711] may be used to provide per message authentication, integrity protection and encryption for both RTP and RTCP streams involved in real-time video and voice sessions. The use of SRTP is recommended for communications over any untrusted network in which confidentiality (or lack of) is a concern. As an example, a voice or video call over a Wi-Fi network (e.g. “Hot Spot”) without any WLAN (Wireless Local Area Network) encryption is highly susceptible to eavesdropping.

The establishment and key exchange for SRTP in RCS shall be based on SDES (Session Description Protocol Security Descriptions for Media Streams, cf. [RFC4568]) which is transported within SDP, following the SIP SDP offer/answer model. SDES and SRTP profiles for media security in IMS are specified in [3GPP TS 33.328].

NOTE: [3GPP TS 33.328] defines two modes of operation for SDES/SRTP: e2ae (end-to-access edge) mode and e2e (end-to-end) mode. For the e2ae mode, SDES is run between an IMS client and a SIP edge proxy, i.e. a P-CSCF (IMS-ALG). An IMS access Gateway controlled by a P-CSCF (IMS-ALG [Application Layer Gateway]) provides the SRTP termination for the “Access Edge”. In the e2e mode, SDES and SRTP is transported end-end between two end user clients.

An RCS client that supports SRTP and SDES and support e2ae mode shall indicate this during the IMS registration according to [3GPP TS 24.229]. The P-CSCF (IMS ALG), if supporting e2ae mode, indicates this to the UE as part of the IMS registration procedures according to [3GPP TS 24.229]. The use of SRTP is enabled through the client configuration parameters (see section A.2.10), and whether it is used or not can be configured differently for Wi-Fi access and cellular access.

However not all end user clients may support SRTP. Therefore, the Service Provider’s network equipment should support e2ae mode. An RCS client that supports SRTP and SDES shall also support e2ae mode.

When using SRTP/SDES, the RCS client can include preference of security mode to use in accordance to [3GPP TS 33.328]. It is recommended that e2ae mode is used by the UE, if also indicated to be supported by the P-CSCF (IMS-ALG). Otherwise, the RCS client may try e2e by not indicating any preference during the session setup.

NOTE: This does not exclude that the Service Provider network still may decide to terminate the media security in the network (P-CSCF (IMS-ALG)).

For terminating sessions, when the UE has indicated support for e2ae SRTP/SDES in the registration, the P-CSCF (IMS-ALG) shall behave as specified in [3GPP TS 24.229], i.e. ensure that SRTP is used, and facilitate interworking from RTP to SRTP when needed.

For terminating session, when the UE has not indicated support for e2ae SRTP/SDES, the P-CSCF (IMS ALG) decides based on local policy, whether to apply SRTP / SDES towards the UE. A possible local policy is that the P-CSCF (IMS-ALG) invokes procedures related to SDP and SRTP for Wi-Fi access, but not for cellular access.

NOTE: Enforcing SRTP/SDES on the terminating call leg towards a UE that does not support SRTP/SDES will lead to the connection establishment failing, which may be an issue for inbound roaming where the operator has no control of what clients are used, or for cases where there are other (non-RCS) clients in the same network that use RTP.

2.13.1.3.2 MSRP

MSRP is used in many RCS services which involve the exchange of images, files and instant messages (e.g. session based). Similar to RTP, MSRP is established through SDP exchanges in SIP signalling and it relies heavily on the security provided in signalling. The use of cryptographically strong random values appended to MSRP URIs exchanged within SDP provides binding between the SIP and MSRP sessions and any identities exchanged within SIP.

For RCS, the use of TLS mode as specified in [RFC4975] is recommended when MSRP is transported over an unsecure network (e.g. Wi-Fi). Consequently, a client configuration parameter to enable Message Session Relay Protocol over Transport Layer Security (MSRPoTLS) is specified in section A.2.10, and whether it is used or not can be configured differently for Wi-Fi access and cellular access.

The RCS client shall use self-signed TLS certificates to produce fingerprints (e.g. secure hash) of the certificate which are exchanged during the SDP negotiation associated with the invitation and MSRP establishment procedure. The certificate fingerprint used for MSRP shall follow the same fingerprint mechanism specified in [RFC4572]. This binding of the certificate fingerprint to SIP signalling relies on the underlying security and trust provided by SIP signalling (e.g. IPsec, SIPoTLS (SIP over TLS), etc.). As a consequence, it is assumed that MSRPoTLS connections shall only happen when combined with the use of encrypted SIP signalling.

When using MSRPoTLS, and with the following two objectives allow compliance with legal interception procedures, the TLS authentication shall be based on self-signed certificates and the MSRP encrypted connection shall be terminated in an element of the Service Provider network providing service to that UE. Mutual authentication shall be applied as defined in [RFC4572].

Since the alternative connection model for MSRP shall be supported as specified in [RFC6135] (see section 2.8) the network will in some cases take the active role, and in some cases take the passive role, in the establishment of the TCP connection. Each peer

(UE and network) shall take the same role (active or passive) in TLS as it took in TCP, so if the network has taken the passive role in TCP, it will also act as TLS server, as specified in [RFC6135]. When TLS is used, both endpoints shall exchange self-signed TLS certificates and fingerprints, as specified in [RFC4572].

In RCS, and in accordance with [RFC4975], all UEs are mandated to support MSRPoTLS as defined in [3GPP TS 24.229-rel12] with certificate fingerprints as defined in [3GPP TS 33.328]. For terminating sessions, the P-CSCF (IMS ALG) decides based on local policy whether to apply MSRPoTLS towards the UE. A possible local policy is that the P-CSCF (IMS-ALG) invokes procedures related to MSRPoTLS for Wi-Fi access, but not for cellular access.

2.13.1.4 XCAP Authentication and Security

XML Configuration Access Protocol (XCAP) exchanges between the RCS client and the XDMS requires authentication and in most cases transport layer security.

The XCAP requests used in RCS shall use the OMA XDM architecture as defined in [XDM2.0_Core]. This means that all XCAP requests from RCS clients need to be authenticated via the Aggregation Proxy (AP) before being forwarded to an XDMS or to a Cross-Network Proxy. The XCAP authentication procedures between the RCS client and the Aggregation Proxy shall be handled as described in section 5.1 of [XDM2.2_Core], with the additional clarifications for RCS listed in this section.

For RCS clients enabled for VoLTE/VoWiFi (see section 2.9.1.1), the XCAP authentication shall be supported as follows:

- When using single registration with VoLTE (i.e. using one single IMS registration for RCS and VoLTE), the same authentication specified for VoLTE use of XCAP, as defined in [PRD-NG.102] shall be supported.
- When the RCS client uses a separate IMS registration from the VoLTE/VoWiFi client, then the XCAP authentication type used may be different than the VoLTE/VoWiFi client's. If it is different, the XCAP authentication method used for RCS shall be HTTP digest based authentication as per section 5.1 of [XDM2.2_Core].
- For RCS clients (and IMS core) that support GAA, the RCS client shall use their AKA credential to fetch an HTTP digest credential using the 3GPP Generic Bootstrapping Architecture (GBA). The RCS client authenticates to an Authentication Proxy (AP) over a TLS secured session using HTTP Digest as per section 5.1 of [XDM2.2_Core].
- For RCS clients (and IMS core) that do not support GAA, the XCAP authentication type shall be HTTP digest. The RCS client shall use its pre-configured credential (e.g. username and password via the XDM configuration parameters in Annex A.1.3.1) to authenticate to the AP over a TLS secured session using HTTP Digest as per section 5.1 of [XDM2.2_Core].

For non-VoLTE/VoWiFi enabled RCS clients, the use of digest authentication over a TLS secured session shall be supported for XCAP authentication to the AP as per section 5.1 of [XDM2.2_Core].

When authentication type used for XCAP is HTTP Digest authentication:

- RCS client may use the same credential (e.g. username and password) as SIP based Digest authentication when applicable.

NOTE: an RCS client may use different authentication types for SIP and XCAP.

- The HTTP digest credentials (e.g. username and password) for XCAP specified in section A.1.3.1 shall be used.

2.13.1.5 Common Message Store Authentication and Security

The RCS client shall support the authentication and security mechanisms described in [RCS-CPM-MSGSTOR-ENDORS] for access to the Common Message Store using IMAP.

Authentication shall be based on username and password stored on the RCS client and the following Plaintext username and password sent using the LOGIN command as specified in [RFC3501].

For SIM-based devices, the user name and password can either be configured through the MESSAGE STORE USER / PASSWORD configuration parameters defined in section A.1.4.3 or be derived through the Generic Bootstrapping Architecture (GBA) as described in section 4.1.4.1. For SIM-less devices, the credentials username and password shall always be provided as part of the configuration.

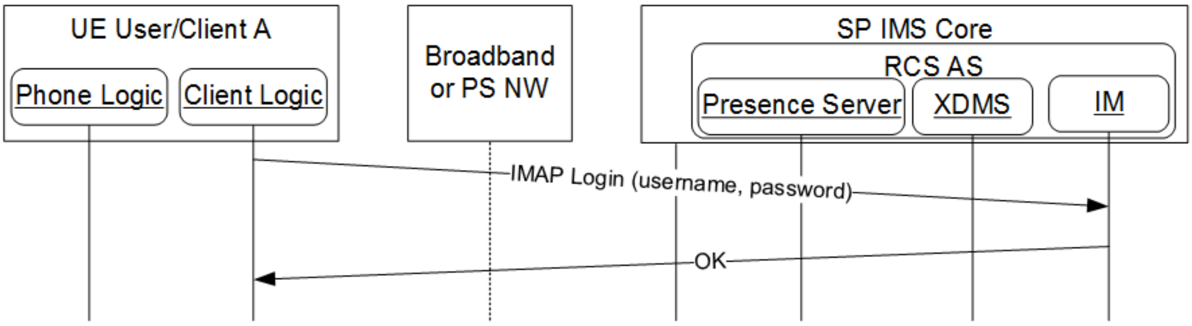

Figure 21: IMAP authentication with SASL or plain text login

TLS shall be used to provide message authentication, integrity protection and confidentiality for the IMAP protocol as specified in [RCS-CPM-MSGSTOR-ENDORS]. TLS must be established before any IMAP based authentication occurs using either the LOGIN or AUTHENTICATE command.

The Common Message Store server shall authenticate itself towards the RCS client using certificate based TLS authentication. The client shall support certificates based on a Public Key Infrastructure (PKI) for which the RCS client is pre-configured with a root or intermediate CA (which is recommended to be a public CA root authority) certificate in the signing chain of the certificate.

2.13.2 Privacy

2.13.2.1 Overview

A key element of promoting user adoption of RCS is gaining the user’s trust with regards to privacy. Service Providers need to provide security mechanisms to ensure unwanted parties cannot gain access to RCS user communications and provide adequate mechanisms to enable users to control the information they share. The key security measures to meet these requirements are outlined in section 2.13.1 and privacy controls are summarised in section 2.13.2.2.

2.13.2.2 Privacy controls

Mechanisms provided in RCS to enable users to control their privacy are identified in this section.

2.13.2.2.1 Multidevice Privacy

Where an RCS user has RCS active across multiple devices this fact shall be obscured from other users.

NOTE: Where an RCS user has RCS active across multiple devices this fact cannot be obscured from devices of other users, since the GRUU and/or sip.instance feature tags shall automatically indicate this fact to these other devices.

2.13.2.2.2 Presence information Privacy

The RCS user shall have the option of controlling who they share their presence information with through a process of accepting, blocking or ignoring an invitation to establish a presence relationship (see section 3.7.4.5).

2.13.2.2.3 Video Privacy

The RCS client shall provide the RCS user with control over when any camera on the device is active.

2.13.2.2.4 Social Presence Information Privacy

The RCS user shall have the option to disable sharing of social presence information.

2.13.2.2.5 Network Address Book Privacy

The Service Provider shall ensure access control to the Network Address Book via a process of authentication.

2.13.2.2.6 Location Privacy

The RCS user shall have the option to control sharing of location information (see section 3.10.1.2).

2.13.2.2.7 Messaging and Chat

An RCS user shall have the option to control information communicated about their actions during messaging communications and chat sessions, including the suppression of “display” notifications and “IsComposing” notifications.

2.14 XDM Handling and Shared XDMS

The support of XDM is an optional functionality for RCS and is only required for Service Providers deploying one of the following services:

- The capability discovery based on presence (see section 2.6.1.2).
- The Personal Network Blacklist (see section 2.15).
- Social Presence (see section 3.7).

2.14.1 Shared XDMS template

Following template shall be used for the *resource-lists* in the Shared XDMS:

Shared XDMS:
AUID: resource-lists
Document name: index
Template:

```
<?xml version="1.0" encoding="UTF-8"?>
<resource-lists xmlns="urn:ietf:params:xml:ns:resource-lists"
                xmlns:xd="urn:oma:xml:xdm:xcap-directory"
                xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <!-- The list oma_buddylist contains references to any individual list used according to OMA IG for presence
  subscriptions. -->
  <list name="oma_buddylist">
    <external anchor="http://xcap.gsma.org/resource-
    lists/users/sip:1234578901@gsma.org/index/~~/resource-lists/list%5B@name=%22rcs%22%5D"/>
    <external anchor="http://xcap.gsma.org/resource-
    lists/users/sip:1234578901@gsma.org/index/~~/resource-
    lists/list%5B@name=%22rcs_basic_spi_only%22%5D"/>
  </list>

  <!-- The list rcs_poll_buddylist contains references to individual lists used for RCS non-VIP Contacts -->
  <list name="rcs_poll_buddylist">
    <external anchor="http://xcap.gsma.org/resource-
    lists/users/sip:1234578901@gsma.org/index/~~/resource-lists/list%5B@name=%22rcs_poll%22%5D"/>
    <external anchor="http://xcap.gsma.org/resource-
    lists/users/sip:1234578901@gsma.org/index/~~/resource-
    lists/list%5B@name=%22rcs_poll_basic_spi_only%22%5D"/>
  </list>

  <!-- The list oma_grantedcontacts contains the list of all granted contacts -->
  <list name="oma_grantedcontacts">
    <external anchor="http://xcap.gsma.org/resource-
    lists/users/sip:1234578901@gsma.org/index/~~/resource-lists/list%5B@name=%22rcs%22%5D"/>
    <external anchor="http://xcap.gsma.org/resource-
    lists/users/sip:1234578901@gsma.org/index/~~/resource-lists/list%5B@name=%22rcs_poll%22%5D"/>
  </list>

  <!-- The list rcs_basic_spi_only_grantedcontacts contains the list of all basic SPI Only granted contacts -->
  <list name="rcs_basic_spi_only_grantedcontacts">
    <external anchor="http://xcap.gsma.org/resource-
    lists/users/sip:1234578901@gsma.org/index/~~/resource-
    lists/list%5B@name=%22rcs_basic_spi_only%22%5D"/>
    <external anchor="http://xcap.gsma.org/resource-
    lists/users/sip:1234578901@gsma.org/index/~~/resource-
    lists/list%5B@name=%22rcs_poll_basic_spi_only%22%5D"/>
  </list>

  <!-- The list oma_blockedcontacts contains the list of all blocked contacts. -->
  <list name="oma_blockedcontacts">
    <external anchor="http://xcap.gsma.org/resource-
    lists/users/sip:1234578901@gsma.org/index/~~/resource-
    lists/list%5B@name=%22rcs_blockedcontacts%22%5D"/>
    <external anchor="http://xcap.gsma.org/resource-
    lists/users/sip:1234578901@gsma.org/index/~~/resource-
    lists/list%5B@name=%22rcs_revokedcontacts%22%5D"/>
```

```
    </list>

    <!-- The list of VIP contacts (buddies) the owner wants to provide all social presence information to. This list
    also includes the owner’s own URI -->
    <list name="rcs">
        <display-name>My presence buddies with location sharing</display-name>
        <entry uri="tel:+1234578901"/>
    </list>

    <!--The list of VIP Contacts (buddies) the owner wants to provide only basic social presence information to-->
    <list name="rcs_basic_spi_only">
        <display-name>My presence buddies without location sharing</display-name>
    </list>

    <!-- The list of NON-VIP Contacts (buddies) the owner wants to provide all social presence information to -->
    <list name="rcs_poll">
        <display-name>My NON-VIP presence contacts with location sharing</display-name>
    </list>

    <!--The list of NON-VIP Contacts (buddies) the owner wants to provide only basic social presence
    information to-->
    <list name="rcs_poll_basic_spi_only">
        <display-name>My NON-VIP presence contacts without location sharing</display-name>
     </list>

    <!-- The list of blocked contacts -->
    <list name="rcs_blockedcontacts">
        <display-name>My blocked contacts</display-name>
    </list>

    <!-- The list of revoked contacts -->
    <list name="rcs_revokedcontacts">
        <display-name>My revoked contacts</display-name>
        <entry uri="tel:+123456" xd:last-modified="2008-12-24T14:32:14Z"/>
    </list>

    <!—PNB lists -->
    <list name="rcs_pnb_chat_blockedusers">
        <display-name>My chat blacklist</display-name>
    </list>
    <list name=" rcs_pnb_ft_blockedusers">
        <display-name> My file transfer blacklist </display-name>
    </list>
    <list name=" rcs_pnb_standalone_blockedusers">
        <display-name> My standalone blacklist </display-name>
    </list>
    <list name=" rcs_pnb_outchat_blockedusers">
        <display-name>My outgoing chat blacklist</display-name>
    </list>
    <list name=" rcs_pnb_outft_blockedusers">
        <display-name> My outgoing file transfer blacklist </display-name>
    </list>
    <list name=" rcs_pnb_outstandalone_blockedusers">
        <display-name>My standalone IM blacklist</display-name>
    </list>
</resource-lists>
```

Table 29: Shared Lists template for RCS

NOTE1: The entry in the "*rcs_revokedcontacts*" list is for illustrative purposes only. It is included as an example since it deviates slightly from the standard list usage. The entry in the "*rcs*" list is also for illustrative purposes only, showing that the user's own URI will be included so the user's clients receive the user's own presence information (see also section 3.7.4.3.3).

NOTE2: The resource-list contains only the lists needed for the features allowed by the service provider (e.g. all the lists related to presence should not be added if only PNB is deployed).

2.14.2 XML Document Handling

When first started the RCS client shall check through a XCAP directory query whether

- The "resource-lists" document exists, if Presence or PNB is deployed by the service provider.
- The "pres-rules", "rls-services" and the "pidf-manipulation" (permanent presence state) documents exist, if Presence is deployed by the service provider.

If they do not exist, the RCS client shall create them if they are applicable (i.e. depending on whether Presence or PNB is deployed as described in the previous bullets). If the documents exist, the RCS client will check whether they comply with the templates defined in sections 2.14.1 and 3.7.4.5.2 by using the following criteria for the documents:

- For the "*resource-lists*" document, first check whether it contains an "*rcs_basic_spi_only*" list. If not, add the "*rcs_basic_spi_only*" and "*rcs_basic_spi_only_grantedcontacts*" lists to the document and modify the "*oma_buddylist*" list to refer to both the "*rcs*" and the "*rcs_basic_spi_only*" lists.
- Secondly check whether it contains an "*rcs_poll_buddylist*" or an "*rcs_poll*" list. If not, add the "*rcs_poll*", "*rcs_poll_basic_spi_only*" and "*rcs_poll_buddylist*" lists to the document and modify the "*oma_grantedcontacts*" list to refer to both the "*rcs*" and the "*rcs_poll*" lists and the "*rcs_basic_spi_only_grantedcontacts*" list to refer to both the "*rcs_basic_spi_only*" and "*rcs_poll_basic_spi_only*" lists.
- For the "*rls-services*" document, firstly check if the "*rcs*" service URI entry refers to the "*oma_buddylist*" list. If the document refers to the "*rcs*" list instead, the RCS client shall modify it to refer to the "*oma_buddylist*" list
- Secondly, check if it contains an "*rcs_poll*" service URI entry. If not, an "*rcs_poll*" service URI entry with a reference to the "*rcs_poll_buddylist*" in Shared XDMS will be added.
- For the "*pres-rules*" document, check whether it contains the "*rcs_basic_spi_only_granted_contacts*" rule. If not, the RCS client shall add this rule to the document.
- For "*rcs_pnb_chat_blockedusers*", "*rcs_pnb_ft_blockedusers*" "*rcs_pnb_standalone_blockedusers*", "*rcs_pnb_outchat_blockedusers*", "*rcs_pnb_outft_blockedusers*" and "*rcs_pnb_outstandalone_blockedusers*" lists in the Shared XDMS, if they do not exist and the PNB MANAGEMENT configuration parameter (see section A.1.3) is set to enabled, they shall be added to the "*resource-lists*" document.

Once the documents have been setup in this way, the RCS client shall only modify the "*rcs*", "*rcs_basic_spi_only*", "*rcs_poll*", "*rcs_poll_basic_spi_only*", "*rcs_revokedcontacts*" and "*rcs_blockedcontacts*", "*rcs_pnb_chat_blockedusers*", "*rcs_pnb_ft_blockedusers*", "*rcs_pnb_standalone_blockedusers*", "*rcs_pnb_outchat_blockedusers*", "*rcs_pnb_outft_blockedusers*" and "*rcs_pnb_outstandalone_blockeduserslists*" in the "*resource-lists*" document. Only if the user explicitly requests to recreate the documents according to the possibility described below, the other documents and parts of the "*resource-lists*" document should be modified.

XDM documents can be updated without the involvement of the RCS client of this RCS release. Two types of changes are possible:

1. Shared lists are updated by adding new entries, removing entries or updating entries.
2. Structural changes to the documents (for example to support new options in the presence authorisation).

In case 1, in order not to overwrite changes done for example by another client, either a conditional update should be done (per XCAP conditional operations as defined in [RFC4825] section 7.11) or the client should retrieve the latest status of the document before doing the update. An RCS client of this RCS release shall support one of these options when updating XDM documents.

Case 2 (structural changes to a XDM document) could occur when an RCS client of this RCS version is deployed in a future RCS environment, even though the future RCS version should be backward compatible with previous ones. The RCS client shall go to a read-only mode with regards to all XDM documents when it detects such changes. Future RCS versions will indicate this by renaming the "*rcs*" shared list. If the list is not renamed, but structural changes were detected in documents in the presence and RLS XDMS, the RCS client will go to read-only mode only for the updated documents. In that case the RCS client indicates to the user that they should use a client with an updated RCS version to carry out commands that require modifying any of such documents.

Circumstances where the user downgrades from a future RCS release to the use of an RCS client only, (for example the end-user does not have a client with an updated RCS version or there is some blocked situation between the XDMC and XDMS), the RCS client shall offer the user the possibility to remove all information stored in the XDMS's, this then creates new documents based on its current status and RCS release. The removal of the documents shall be based on a retrieval of the complete list of documents using XCAP Directory requests and then removing all listed documents (thus including documents unknown to the RCS client of this RCS release) using relevant operation such as XCAP PUT/DELETE.

Should a device for its own internal use maintain a local copy of the Shared XDMS's "*resource-lists*" document (see section 2.14.1) or the information contained therein, then it shall verify with the Shared XDMS whether its copy is still up to date in the following situations:

- When the client comes online
- When it receives a notification within the dialog of its RLS subscription indicating that the subscription to a contact is pending or active and according to the locally maintained information, it is not aware that the user is part of the RCS buddy list.

NOTE1: This situation can occur, when the user invites the contact to share social presence information from another client, or a contact has been added as a VIP-contact from another client.

- When it receives a notification within the dialog of its watcher information subscription indicating that a subscription from a contact changed from the “pending” to the “active” or “terminated” state when no action was taken to authorise or block that subscription from the client. The state change to “terminated” should only be taken into account for this case when the event triggering the state change indicates “rejected”.

NOTE2: This situation can occur when the user authorises or blocks the subscription from another client.

- When it receives a notification within the dialog of its RLS subscription indicating that the subscription to a contact that is presence enabled was terminated with reason “timeout” when no action was taken from the client to revoke the presence sharing with that contact.
- When it receives a notification within the dialog of its RLS subscription indicating that the subscription to a contact that is presence enabled was terminated with reason “noresource” when no action was taken from the client with that contact.

NOTE3: This situation can occur, when the user changes a contact from being a VIP contact to being a non-VIP contact from another client.

NOTE4: A device is not required to maintain a local copy of the Shared XDMS’s “*resource-lists*” document. If it does not, for presence it can simply display the presence information it receives and it does not need to access the XDMS.

2.14.2.1 Client XML procedures and multi-device

The XML Document Management Client (XDMC) from the RCS client performs the XCAP Get and Put operations on the Shared List XDMS *resource-lists*. Once updated by the user, the RCS client shall store the updated version of the document in the Shared List XDMS for further access from his other devices.

The RCS client shall not cache the document and fetch the latest version of the document from the XDMS prior to displaying the list to the user or enabling them to make modifications in it. The RCS client shall not subscribe to the updates of the document as described in [XDM2.0_Core].

2.14.2.2 Authorising XCAP Requests

XCAP requests need to be authorised by the XDMS. This authorisation relies on an assertion of the identity of the requestor of an XCAP request.

The HTTP header fields *X-XCAP-Asserted-Identity* and *X-3GPP-Asserted-Identity* used to contain the asserted identity of a requestor of an XCAP request may depend on operational conditions (type of access used by the terminal, Service Provider policy) for example different Service Providers may apply different algorithms to assert the identity of a requestor of an XCAP request. Thus, for any Authorisation check to be carried out by the XDMS, any of both *X-XCAP-Asserted-Identity* and *X-3GPP-Asserted-Identity* header fields are accepted as a valid header field containing the asserted identity of the requestor of the XCAP request inside the Service Provider domain.

To offer a unique inter-Service Provider interface, the *X-3GPP-Asserted-Identity* header field is always conveyed between two Service Provider domains, at the NNI interface.

When the terminal of a watcher requests, via XCAP, some content (for example status-icon, refer to section 3.7.4.4.2.3) associated with the presence document of a presentity, the XDMS of the presentity has to check whether the watcher is authorised to access this content, according to the presentity's presence subscription rules.

As defined in sections 2.14.1 and 3.7.4.5.2, amongst others the "*rcs*" list is granted this permission.

The lists in section 2.14.1 can contain both SIP URI and tel URI address of authorised watchers in a Service Provider domain. To ensure both cases at the NNI interface, the “*X-3GPP-Asserted-Identity*“ of the initiator of an XCAP request should contain both the sip URI and tel URI of this user.

2.15 Personal Network Blacklists (PNB)

With this optional RCS feature that is enabled using the PNB MANAGEMENT configuration parameter (see section A.1.3), the RCS user may be provided with the possibility to manage their Personal Network Blacklist (PNB), in order to either prevent receiving undesired communications, messages or media.

The PNB feature relies on the Shared XDMS that is also used for SPI (see section 3.7). New lists are pre-defined in the Shared List XDMS as follows:

- “*rcs_pnb_chat_blockedusers*”: this list contains all blocked senders for chat
- “*rcs_pnb_ft_blockedusers*”: this list contains all blocked senders for file transfer
- “*rcs_pnb_standalone_blockedusers*”: this list contains all blocked senders for standalone messages
- “*rcs_pnb_outchat_blockedusers*”: this list contains all blocked recipients for chat
- “*rcs_pnb_outft_blockedusers*”: this list contains all blocked recipients for file transfer
- “*rcs_pnb_outstandalone_blockedusers*”: this list contains all blocked recipients for standalone messages

2.15.1 RCS Applicability

The enforcement of the PNB feature is performed by the Blacklist Policy Enforcement Function (BPEF) that could both:

- Be implemented as part of the relevant RCS application server (e.g. RCS Messaging Server), or,
- In a separate server, which enforces the policy.

The PNB feature applies only to the following RCS services:

- Standalone messaging (see section 3.2)
- 1-to-1 chat (see section 3.3)
- File transfer (see section 3.5).

Taking into account the supported features, the BPEF can be completely collocated with the Messaging Server.

The PNB feature can be enabled or disabled. When enabled:

- The BPEF shall apply triggers for checking the RCS user's PNB, during the relevant RCS service traffic, on both originating and terminating traffic.
- RCS clients shall be configured with the PNB MANAGEMENT parameter (described in Annex A, section A.1.3), so it is possible to handle the PNB configuration from the client.

The functionality provided by BPEF is summarized below:

- On originating side:
    - Checks the outgoing blacklists for the respective request (i.e. chat, file transfer or standalone messaging) and if the recipient is found in the applicable list (e.g. rcs_pnb_outchat_blockedusers for chat), the request is rejected.
- On the terminating side:
    - Checks the blacklists for the respective request (i.e. chat, file transfer or standalone messaging) and if the sender is found on the pertaining list (e.g. chat_blockedusers for chat), the request is rejected.
    - If supporting the Common Message Store feature and if allowed by local server policy, the Messaging Server stores the blocked message/chat/File Transfer event with the metadata content following Conversation History format as per [RCS-CPM-MSGSTOR-ENDORS].

2.15.2 PNB management

The PNB can be updated in any of the following ways:

1. The user performs the management of the lists from their RCS client by adding or deleting users in the list entries of the Shared List XDMS using XCAP, or
2. The RCS client supports the user, by prompting him/her to choose upon a rejection (e.g. chat, group chat or file transfer) to add the originator in the respective blacklist.

Once the user rejected a Chat invitation, or a File Transfer, the RCS client UI may prompt them to select whether the rejection is permanent, or if it is a onetime rejection. A permanent rejection will trigger an update in the user's respective black list, by adding the originator in that list.

Other options may be possible, such as allowing the RCS user to add other users in their PNBs from the viewing of the Conversation History.

Regardless of the trigger for the PNB update by the RCS Client, when such update needs to be done in the network, the RCS Client shall issue an XCAP request as per [XDM2.0_Core] procedures and as described in section 2.14.2.

2.15.2.1 Authorising PNB management requests

The authorisation of the PNB management requests (get, update, delete) is done as described in section 2.14.2.2.

2.16 Emergency Services

2.16.1 General

In some markets, regulatory requirements are emerging for IMS Multimedia Emergency Services. UEs and the network in required markets must support the 3GPP Release 11 IMS Emergency Services as specified in [3GPP TS 24.229-rel11], [3GPP TS 23.167], Chapter 6 and Annex H, and 3GPP Release 11 emergency procedures specified in [3GPP TS 24.301].

Please note [PRD-IR.92] in section 5.2 and [PRD-IR.51] in section 5.3specify Emergency Services support.

2.16.2 RCS Service Feature List

Emergency Services support is provided in the following RCS Service Feature:

- 1-to-1 Chat

3 RCS Services

3.1 General Service Overview

RCS provides several services that fit into the framework defined in section 2. As mentioned in section 1.2 all of these services are optional for a Service Provider to deploy.

The first set of services is intended to enhance the user's messaging experience. Section 3.2 describes the standalone messaging service based on OMA CPM that is considered as an evolution of the SMS/MMS messaging services providing fewer restrictions and provides the interworking capability with those services. Section 3.3 introduces the 1-to-1 chat service that provides a more real-time experience through "IsComposing" indications next to the store and forward functionality, including delivery and display notifications, that allows reaching users while they are offline. In section 3.4, it is described how the 1-to-1 chat service is extended to multiparty scenarios. For both the 1-to-1 chat and for this Group Chat, the technical realisation can be based on either OMA SIMPLE IM or OMA CPM. Interworking between these realisations has been described to manage these as a single service providing transparency and an enhancement to the UX.

As a service that is closely related to the messaging in that it is used for the exchange of discrete content and is based on the same underlying technology, chapter 3.5 describes the File Transfer service allowing a user to exchange any type of file with another user.

Chapter 3.6 introduces the content sharing services allowing the user to share multi-media content ahead of the call to provide context to the called party when the call is set up. The content sharing services also includes the capability to exchange a video, an image, a map or a drawing canvas in real-time with another user during a voice call. In other circumstances the File Transfer service or the messaging service could be used. It also allows to share a note (reason) or a voice message after an unanswered call.

The social presence service in chapter 3.7 allows the user to announce a status including a picture, a link and possibly even information related to his location to a subset of his contacts while at the same time receiving status updates from those same contacts. Depending on the user's preference regarding a contact, they could be informed about such status changes in real-time or after a potentially long delay.

Section 3.8 and 3.9 describe respectively an IP based voice and video call functionality for broadband access clients and mobile devices. These services include support for a set of supplementary services and ensure the quality of service delivery when used on EPC-integrated Wi-Fi and LTE access. For the voice call, a mobile device on EPC-integrated Wi-Fi and LTE provides continuity to a CS call if network coverage circumstances require this. These services are based on [PRD-IR.51] and [PRD-IR.92] for the voice call and [PRD-IR.94] for the video call.

A geolocation service is introduced in section 3.10 which allows a user to share their location (or any other desired location) with a contact including requesting the location of a contact.

All these services can be invoked either from within the address book provided that the contact has the corresponding capability (see section 2.6) and the current network connectivity allows using the service (see section 2.6.4.1) or directly from the device's menu. Additional entry points may be the chat and call history, the media gallery and camera application depending on what is suitable for the service.

Most of the NNI handling is done as described in section 2.12.

3.2 Standalone messaging

RCS provides a Standalone Messaging service as described in [RCS-CPM-CONVFUNC-ENDORS]. It includes both text and multi-media messaging services using IMS-based OMA CPM Standalone Messaging instead of the SMS and the MMS.

The use of OMA CPM Standalone Messaging removes some of the limitations associated with a messaging service deployment based on the SMS and MMS services, e.g., the 160-character message size, content type, lack of display notifications for text messages and support for the service users with multiple devices.

In addition, the RCS Standalone Messaging service supports interworking to SMS and MMS as described in [RCS-CPM-IW-ENDORS].

A conversational view of the CPM standalone messaging is used in RCS.

3.2.1 Feature description

The feature list of the RCS standalone messaging service includes the following main features:

- Standalone messaging (text and multimedia)
- Delivery and Display Notifications
- Support for multiple devices per user
- Deferred Messaging
- Support for the Common Message Store
- Interworking with legacy messaging services

These features are further described in sections below.

3.2.1.1 Standalone messaging

The RCS standalone messaging capability employs the OMA CPM's SIP-based standalone messaging as described in [RCS-CPM-CONVFUNC-ENDORS]. It evolves the two separate text and multimedia messaging mechanisms into one single and unified messaging framework. This converged messaging mechanism uses the combination of the Pager Mode messaging mechanism and the Large Message Mode messaging mechanism. The mode is selected based on the message size. Smaller messages are sent via Pager Mode and larger messages via Large Message Mode. This built-in capability of the RCS Standalone Messaging enhances the user experience by making the selection transparent to the user: the user does not have to choose between messaging technologies based on either the media type or artificially imposed size limits. In addition, the RCS Standalone Messaging further facilitates the transition from the currently distinct SMS and MMS messaging services towards a single all-IP Messaging services.

The RCS standalone messaging includes support for the following specific features:

1. In supporting both text and multi-media messaging, it does not make a distinction between text and multimedia messages.
2. Its' message delivery includes both 1-to-1 and group messaging including support for "reply-to-all" functionality.
3. Imposes no limitations on the message size and media types. However, the maximum message size can be controlled by Service Providers.
4. Capabilities for both broadband access and mobile access terminals.
5. It can store a message exchange both in a local and a Common Message Store and to present a conversational view of the exchanged messages.
6. Provides message delivery and display notifications.

3.2.1.2 Delivery and display notifications

Upon sending an RCS Standalone Message including a request for message disposition state, the sender shall receive a delivery notification and may receive a display notification.

If an RCS Standalone Message contains a disposition notification request targeted at a group of recipients or when multiple disposition notifications are expected to arrive for the same standalone message, the originating user may receive aggregated disposition notifications based on Service Provider policies. Aggregating disposition notifications may be performed by the originating Participating Function or the Controlling Function.

In the case of delivering an RCS Standalone Message to multiple devices of the same contact/user, the terminating Participating Function shall, for each disposition notification type (i.e. delivery and display notifications), forward the first disposition notification received to the originator of the message and shall suppress the forwarding of subsequent disposition notifications received from the other devices that the message was delivered to.

3.2.1.3 Support for multiple user devices

The RCS standalone messaging supports users with multiple devices. The RCS standalone messaging service shall be available on all of the RCS capable devices/clients of a user. More specifically, an incoming message shall be delivered to all clients of a user, which are online and capable of handling the RCS standalone messaging service. If all clients of a user were offline when a message has to be delivered, the message will be delivered to the first client that comes online if the message has not expired in the meantime. The procedures for handling the multiple devices are described in [RCS-CPM-CONVFUNC-ENDORS].

3.2.1.4 Deferred Messaging

As opposed to immediate message delivery, the RCS "deferred messaging" is to temporarily hold the message in the terminating Participating Function and deliver it at a later time. Furthermore, the deferred messaging is to defer the delivery of standalone messages when none of the terminating RCS user's devices is registered and available to receive the messages. In this case, the undelivered messages stay in the RCS Participating Function until they are either delivered to the user devices, are deleted or expire. The procedures for handling the deferred standalone messages are described in [RCS-CPM-CONVFUNC-ENDORS].

3.2.1.5 Interworking with legacy messaging services

The purpose of this feature on interworking between the RCS standalone messaging and the legacy messaging services, e.g., SMS, MMS, is to communicate, in a seamless manner, with devices or networks that support legacy SMS and/or MMS messaging services.

3.2.1.6 Personal Network Blacklists handling

NOTE: In the present section, it is assumed that the BPEF as described in section 2.15.1 is provided by the Messaging Server.

When supported, the user defined Personal Network Blacklists are applied by the Messaging Server at both origination and termination of standalone messages.

If any of the recipients of a Standalone message are found in the corresponding Standalone blacklists at either origination and/or termination, the Messaging Server:

- At the originating side:
  - Removes the recipient at origination, and continues with processing the message to the remaining recipient(s). If the recipient is the only recipient of the message, then the message is discarded and an error is returned to the originator user.
- On the terminating side:
  - Stores the message(s) blocking event in the Blocked Folder.
  - Checks whether a notification should be sent to the user about the blocked message or not, based on Service Provider policies.

3.2.2 Interaction with other RCS features

When RCS VV-Mail is deployed, Standalone Messaging service is used for new message notification to RCS clients with the RCS VV-Mail specific feature tag *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.vvm"*. In this case the interworking requires operator specific handling. See RCS VV-Mail section 3.13 for further details.

3.2.3 High Level Requirements

This section contains Standalone Messaging service's high level requirements. These requirements are listed in two separate support aspects for client and server as follows:

3.2.3.1 Client/device support

3-2-1 Delivery and Display Notifications: Supporting RCS user to request and receive notifications on the disposition state of a standalone message they have sent. Furthermore, the client device should allow both the sending and receiving users to optionally enable/disable the display notifications request and response, respectively.

3.2.3.2 Server support

3-2-2 Number of recipients: For the Standalone Messaging to support both 1-to-1 and 1-to-many (group) messaging features including "reply-to-all" for the group messaging.

3-2-3 Multiple clients/devices: The ability to support RCS users employing multiple RCS capable devices/clients.

3-2-4 Interworking with legacy SMS and/or MMS: The ability to interwork and communicate with other messaging servers supporting legacy SMS and/or MMS messaging services.
3-2-5 Deferred messaging: To defer the delivery of an RCS Standalone Message when none of the terminating RCS devices is registered and available to receive the RCS Standalone Message.
3-2-6 Delivery and Display Notifications: The server shall ensure that requests for disposition notifications and the notifications themselves are delivered correctly.

3.2.4 Technical Realisation

3.2.4.1 Standalone messaging

The technical realisation of the RCS standalone messaging is based on the OMA CPM Pager Mode and Large Message Mode mechanisms as described in [RCS-CPM-CONVFUNC-ENDORS]. These messaging modes in conjunction with the 3GPP IMS functional entities as the infra-structure for the messaging functional entities are used as the platform for providing an end-to-end standalone messaging service.

Both CPM Pager Mode and Large Message Mode Standalone Messaging mechanisms are based on the use of the IETF SIP protocol. The Pager Mode messaging uses the SIP MESSAGE method, which imposes a limitation for the maximum message size, while the Large Message Mode messaging uses dedicated SIP/MSRP sessions set up for the delivery of large messages without limiting the message size.

The maximum size of an RCS Standalone Message to be sent using the Pager Mode messaging cannot exceed 1300 bytes. Messages with size exceeding this threshold will be handled by the Large Message Mode messaging. Therefore, an RCS Standalone Message will be sent and delivered using either the Pager Mode or the Large Message Mode depending on the size of the message. This procedure is transparent to the user, i.e., the user does not make the decision to use either Pager Mode or Large Message Mode messaging nor do they see a difference in the service behaviour.

From the user access perspective, the same technology is used for simultaneous delivery to mobile and broadband access clients.

3.2.4.1.1 Pager Mode Messaging

Figure 22 presents an architectural view of the RCS standalone messaging employing Pager Mode messaging.

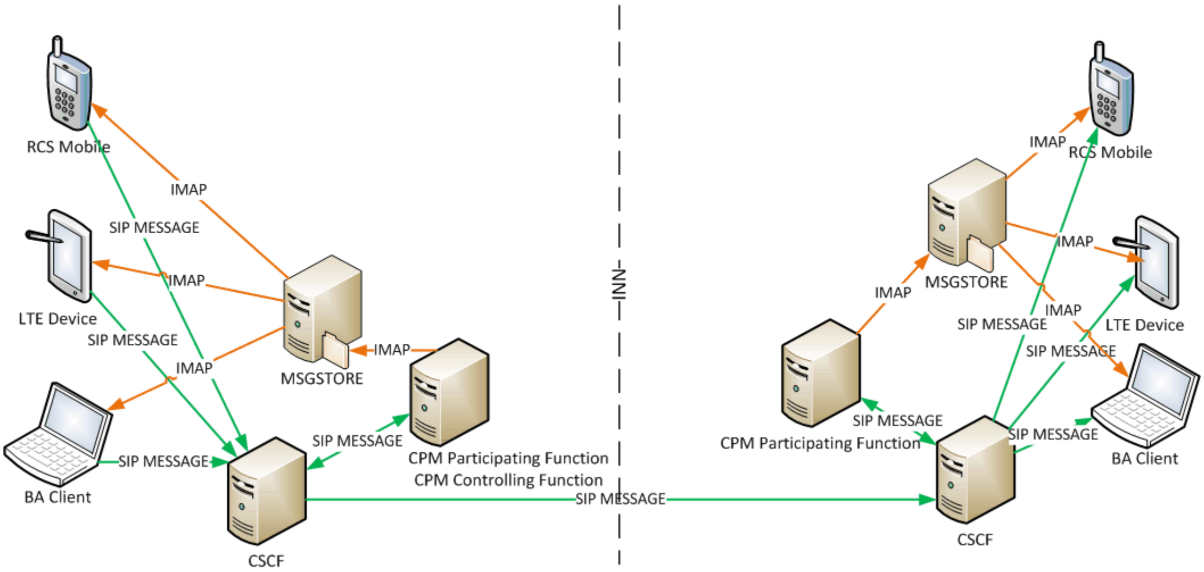

Figure 22: Standalone Messaging using Pager Mode

The detailed procedures for the sending and delivering of a message to the recipient are described in [RCS-CPM-CONVFUNC-ENDORS]. From the sending user client/device, the message will pass through the Participating Functions at the originating and terminating sides to be delivered to the intended receiving client(s).

If the message is targeted for a group of recipient users, it will be sent from the Participating Function in the originating side to a Controlling Function, also in the originating side that will then perform the procedures for distributing the message to the Participating Functions attending the intended recipient clients.

The RCS Standalone Message delivery and display notifications will follow the reverse path that was used for sending the message.

As described in [RCS-CPM-CONVFUNC-ENDORS], if the Common Message Store is provided any standalone message that is sent or received will be stored in the corresponding RCS user's Message Store as described in Section 4.1.

3.2.4.1.2 Large Message Mode Messaging

Figure 23 presents an architectural view of the RCS Standalone Messaging employing the Large Message Mode messaging.

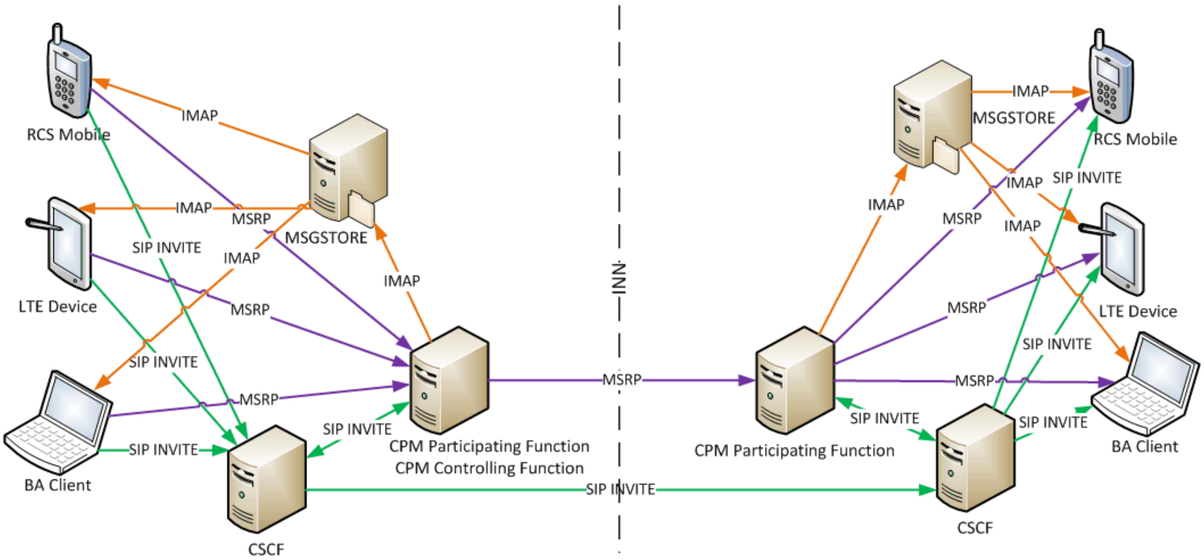

Figure 23: Standalone messaging using Large Message Mode

A large text or multimedia message is sent from an RCS client and delivered to the target client using the Large Message Mode messaging mechanism as described in [RCS-CPM-CONVFUNC-ENDORS]. Through an MSRP session established following a successful SIP INVITE, the message will be passed through the Participating Functions in the originating and terminating sides to reach the intended recipient. The SIP INVITE includes the size of the standalone message and the content type(s) used in the message.

The terminating Participating Function, amongst other procedures, performs the procedures for deferring messages if none of the intended recipient's RCS capable devices is online.

If the message is targeted for a group of recipient users, it will pass through the Participating Function in the originating side to the Controlling Function also at the originating side before reaching the Participating Function(s) at the terminating side(s). The Controlling Function handles the distribution of the message to various target recipients. As in this case, a list of recipients will be provided along with the delivered message, each recipient has the possibility to send a reply to the sender as well as to all the other users that were addressed in the original message.

The delivery and display notifications of a sent standalone message will follow the reverse path of the sent message.

As described in [RCS-CPM-CONVFUNC-ENDORS], if the Common Message Store is provided any standalone message that is sent or received will be stored in the corresponding RCS user's Network-based Common Message Store as described in section 4.1.

3.2.4.1.3 Standalone Messaging Service identification

The RCS client shall populate the P-Preferred-Service header field in all CPM requests with the CPM Feature tag defined for the service, as described in [RCS-CPM-CONVFUNC-ENDORS]. The S-CSCF or AS that performs the service assertion in the originating network shall add the P-Asserted-Service header field set to the value of the asserted CPM service ICSI (i.e. standalone messaging, such as: "*urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg*" for Pager Mode, "*urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.group*" for Pager Mode to a group, or "*urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg*" for Large Message Mode, or "*urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group*" for Large Message Mode to a group or "*urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred*" for deferred delivery) and remove the P-Preferred-Service header field before further routing the request.

A receiving network element and RCS client should ignore any SIP header fields that they do not understand (e.g. P-Preferred-Service, or P-Asserted-Service header fields).

3.2.4.2 Delivery and Display Notifications

The disposition status notifications for a sent standalone message will follow the reverse path of the sent message. The disposition notifications for the standalone messaging could be used for the 1-to-1 or 1-to-many messaging and for two types of notifications, delivery and display, as specified in [RCS-CPM-CONVFUNC-ENDORS].

For network optimisation purposes, the aggregation of IMDNs as specified in [RFC5438] may be supported for network initiated IMDNs:

- Within the Service Provider's own network, the aggregation of IMDN may be supported (per local policy).
- For inter-Service Provider interoperability, the individual IMDN will always be sent to the target network, where the aggregation of IMDN is up to the target network (per local policy). That is, if the aggregated IMDNs received by the Messaging Server contain IMDNs that need to be sent to another network, the Messaging Server will repackage the aggregated IMDNs accordingly before sending them to the Chat message sender on the other network.
- If the aggregated IMDNs received by the Messaging Server contain both in-network and inter-Service Provider Chat message senders, the Messaging Server will repackage the aggregated IMDNs according to in-network Chat message senders and inter-Service Provider Chat Message senders.

3.2.4.3 Deferred Messaging

The terminating Participating Function, amongst other procedures, performs the procedure for deferring messages if none of the RCS capable devices of the recipient is online.

When no RCS target recipient client is registered, the terminating Participating Function holding the message for delivery may decide to defer the standalone message for delivery at a later time. For the delivery of a deferred standalone message, the Participating Function has the following options as specified in [RCS-CPM-CONVFUNC-ENDORS]:

1. To send a notification to the RCS clients of the target recipient and wait for these client(s) to take action,
2. To push the deferred standalone messages once one of the clients of the target recipient RCS user becomes available.

NOTE: Service provider's policies may guide which option to adopt.

If a deferred standalone message expires before it is delivered, the terminating Participating Function shall handle the deferred message by discarding it.

3.2.4.4 Personal Network Blacklists handling

NOTE: In the present section, the BPEF as described in section 2.15.1 may be provided by the Messaging Server.

When supported, the PNBs are applied by the BPEF at both origination and termination of standalone messages.

The following resource-lists from Shared XDMS (see section 2.14.1) are checked by the BPEF by comparing the URI values used in the request and in the list:

- At Standalone message origination:

  a) The BPEF checks the 'rcs_pnb_outstandalone_blockedusers' list to verify that the recipient(s) is/are not among the blocked users for this request by comparing URIs contained in the list with the URI value of the Request URI of the SIP request for a 1-to-1 message or with the URIs in the recipient-list body for 1-to-many message.
  b) If true, the BPEF:

     - Removes the recipient from the list of recipients before continuing to process and sending out the Standalone message;
     - If the recipient is the only one in the message, then the message is discarded and a *403 Forbidden* response with a warning header set to “122 Function not allowed” is returned to the user.
     - If there are multiple recipients of the message, the number of acceptable recipients is checked by the Messaging Server after the Personal Network Blacklists verification.

- On termination, the BPEF checks the 'rcs_pnb_standalone_blockedusers' list, to verify if the sender of the Standalone message is among the blacklisted users by comparing the URIs contained in the list with the URI values of the P-Asserted-Identity header field of the SIP request.

  a) If true, the BPEF:

     - Shall return a *403 Forbidden* response with a warning header set to “122 Function not allowed” and,
     - It suppresses any further IM notifications (“delivered” and/or “displayed”) for the blocked messages, and
     - The BPEF stores received blocking event in the dedicated Blocked Folder.

3.2.4.5 Multidevice handling

The RCS supports delivering of standalone messages to multiple devices. As described in [RCS-CPM-CONVFUNC-ENDORS], the delivery of Standalone messages will be done to all the user's RCS devices that are online. Also, when applicable, the message is delivered to a single non-RCS device of the user through interworking with either SMS or MMS as explained in Section 3.2.4.6.

The support of the RCS multidevice environment includes the following major features:

1. When a user sends a message from one of their devices capable of handling the RCS standalone messaging and a Common Message Store is available, all other online devices capable of handling the RCS standalone messaging services shall display the message along with related information such as message state and its disposition.
2. If a Common Message Store is available, all offline clients supporting the RCS standalone messaging service will be capable of showing the messages that the user has sent and received (except for already deleted messages) when the clients are back online.
3. Handling of delivery and display notifications when multiple clients receive a message, the terminating RCS Participating Function shall support forwarding both delivery and display notifications to the originating client, by forwarding the first disposition notification received from one of the devices that the standalone message was delivered to. It suppresses forwarding subsequent disposition notifications received from the other devices to which the message was delivered.

All procedures for sending and receiving standalone messages and their disposition notifications in an RCS multidevice environment, where the RCS user employs multiple devices, are performed as described in [CPM-SYS_DESC] and specified in [RCS-CPM-CONVFUNC-ENDORS]

3.2.4.6 Interworking with Legacy Messaging services

The [RCS-CPM-IW-ENDORS] document describes general interworking procedures applicable to both SMS and MMS and the realisation details for the SMS and MMS interworking. The interworking procedures for the SMS include references to 3GPP's IP-SM-GW (IP Short Message Gateway) as described in [RCS-3GPP-SMSIW-ENDORS].

3.2.4.6.1 Interworking procedure

The procedures for the RCS standalone messaging service feature interworking to SMS and MMS legacy messaging services are performed by two interworking functional entities, the Interworking Selection Function (ISF) and the Interworking Function (IWF). After the Participating Function has decided that the message has to be interworked the selection of whether to interwork to SMS or MMS is done in the ISF as described in [RCS-CPM-IW-ENDORS]. The actual interworking procedure is performed by the SMS and MMS gateways described in [RCS-3GPP-SMSIW-ENDORS] and [RCS-CPM-IW-ENDORS]. These functions also interwork the delivery notifications received from the SMS and the delivery and display notifications received from the MMS message recipient(s) and forward them to the sending Participating Function to be passed on to the sending RCS client.

The interworking functions also interwork any incoming SMS or MMS messages to RCS messaging.

3.2.4.6.2 Interworking with SMS

When the target recipient device for an RCS Standalone Message is a non-RCS capable, an SMS capable device, the process of interworking with legacy SMS is invoked according to [RCS5-CPM-CONVFUNC-ENDORS]. In Figure 24, an architectural view of the RCS standalone messaging service interworking with the legacy SMS is shown. The legacy mobile device is shown as a non-RCS device.

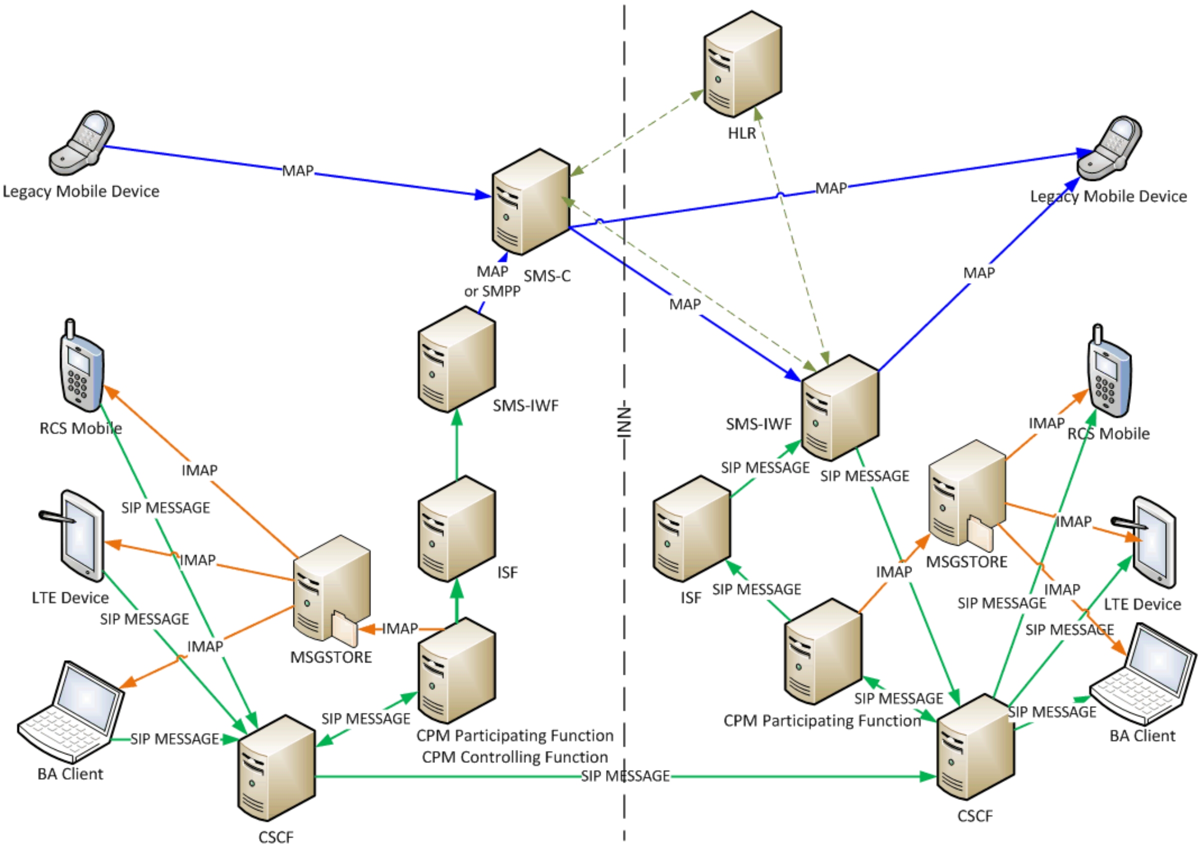

Figure 24: Standalone Messaging interworking with SMS

When the SMS interworking function (IP-SM-GW or SMS-IWF) receives a SIP MESSAGE request with the OMA CPM ICSI "*3gpp-service.ims.icsi.oma.cpm.msg*", it checks the size of the received payload of the SIP MESSAGE request. If the size of the payload is too large to be sent as one SMS message, the payload will be divided into concatenated SMS messages. The SMS-IWF will send the request(s) generated based on the received SIP MESSAGE request towards the SMS-C (Short Message Service Centre) using either the SMPP (Short Message Peer-to-Peer) or MAP (Mobile Application Part) protocols, depending on the type of SMS network in which it is deployed, as specified in [RCS-CPM-IW-ENDORS] or [RCS-3GPP-SMSIW-ENDORS] respectively.

NOTE: For clarity, Figure 24 mainly shows the latter deployment option since the differences between both options are in the existing SMS deployments and therefore have no impact on the Standalone Messaging service.

Breakout to SMS can be done at the originating side if the addressed user is not an IMS user. This is determined based on the standalone messaging capability information, on local information the Messaging Server may have about the recipient, or when the Messaging Server receives an error response. Otherwise, the breakout at the terminating side is done, if either the addressed user is an RCS user using SMS instead of RCS standalone messaging service or the user is using a mixture of legacy and RCS devices.

The following error responses to the SIP MESSAGE (or, for the IP-SM-GW realisation, optionally for a Large Message Mode message the SIP INVITE) request indicate that the recipient is not an RCS contact and these responses can be used to trigger interworking:

- 404 Not Found;
- 405 Method Not Allowed;
- 410 Gone;
- 414 Request URI Too Long;
- 415 Unsupported Media Type;
- 416 Unsupported URI Scheme;
- 488 Not Acceptable Here;
- 606 Not Acceptable.

The case for delivering text messages to a (primary) broadband client of a non-Standalone Messaging user is beyond the SMS interworking gateway of the standalone messaging and its platform. It is not shown in Figure 24 to avoid overloading it. In that scenario the MAP (Mobile Application Part) or SMPP request from the SMS-IWF to the legacy Mobile Device for the incoming SMS message would be replaced by a SMSoIP (SMS over IP) request, which is relayed to the legacy BA Client via the Serving Call Session Control Function (S-CSCF).

3.2.4.6.3 Interworking with MMS

When the target recipient device for standalone messaging is not an RCS device and the message to be sent is a multimedia message, the process of interworking with legacy MMS is invoked according to [RCS-CPM-CONVFUNC-ENDORS].

Figure 25 presents an architecture view of the interworking for multimedia messaging. As shown, the legacy mobile device at the terminating side may either be an RCS user’s primary device that uses MMS instead of RCS Standalone Messaging or a non-RCS device capable of receiving MMS.

Depending on the size of the standalone message, it could be either a text message with a large payload or a multi-media standalone message. In the former case the interworking with SMS would apply as described in section 3.2.4.6.2 if the message were small enough for a concatenated SMS. Otherwise, the interworking would be to the MMS service, hence sending a SIP INVITE request to the RCS MMS-IWF.

When the RCS MMS-IWF receives a SIP INVITE request containing the OMA CPM ICSI “*3gpp-service.ims.icsi.oma.cpm.largemsg*” for a Large Message Mode standalone message, it will send a 200 "OK" response if no errors are found in the SIP INVITE request or an appropriate error response. This is followed by the MMS-IWF’s subsequent receiving of an MSRP SEND request for the establishment of the MSRP session, and the process then continues as described in [RCS-CPM-IW-ENDORS].

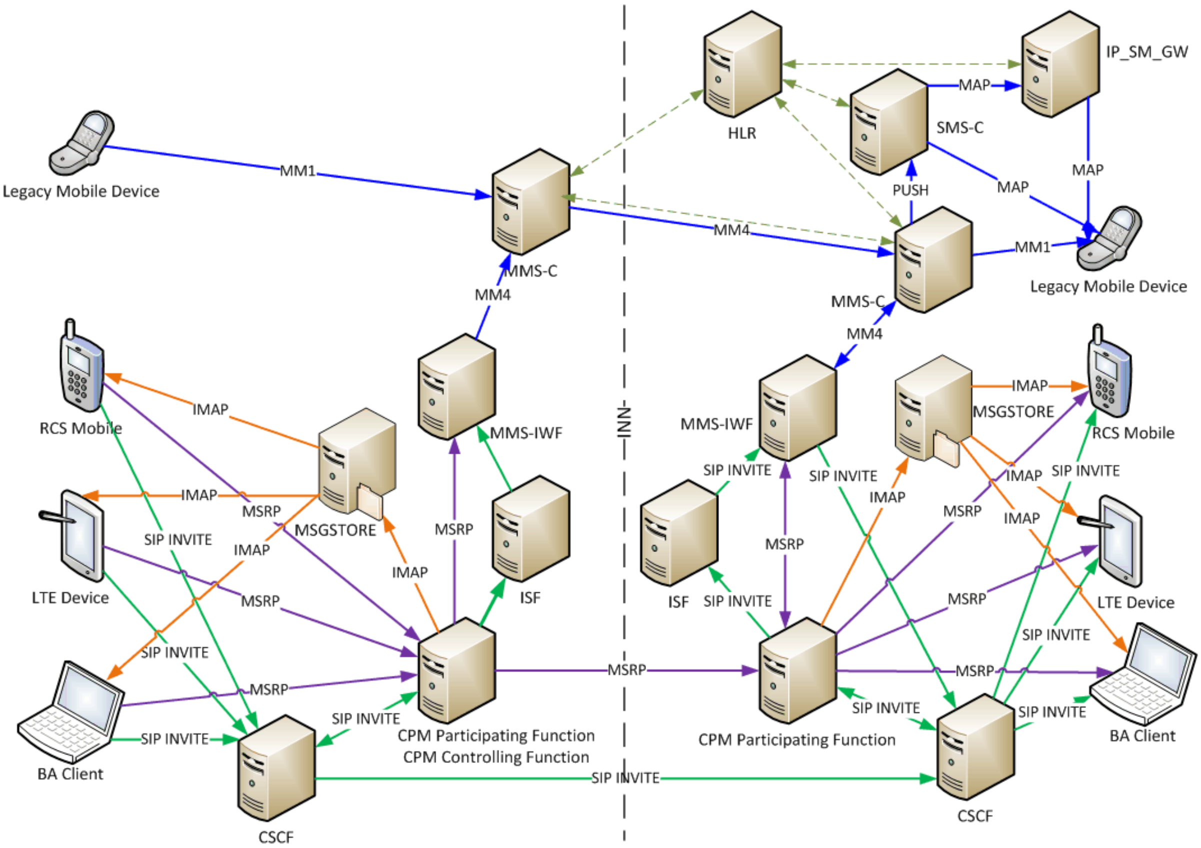

Figure 25: Standalone messaging interworking with MMS

Breakout to MMS can be done at the originating side if the addressee is not an IMS user either based on local information the Messaging Server may have about the recipient, or when it receives an error response. Otherwise, the breakout at the terminating side is done if either the addressee is an RCS user using MMS instead of RCS standalone messaging service or the user is using a mixture of legacy and RCS devices.

The following error responses to the INVITE request indicate the recipient is not an RCS contact and can be used to trigger interworking:

- 404 Not Found;
- 405 Method Not Allowed;
- 410 Gone;
- 414 Request URI Too Long;
- 415 Unsupported Media Type;
- 416 Unsupported URI Scheme;
- 488 Not Acceptable Here;
- 606 Not Acceptable.

Similar to SMS interworking, in the MMS interworking of Figure 25, the case for delivery of the multimedia message to a Broadband Access client of a non-RCS user is not shown to not overload the figure.

3.2.5 NNI and IOT considerations

For the Standalone Messaging service three NNI interfaces are possible:

- The NNI described in section 2.12 carrying the standalone messaging service across RCS compliant networks
- SMS NNI
- MMS NNI

Which of these interfaces is used is decided based on the Service Provider's policies and the applicable interworking agreements.

3.2.6 Implementation guidelines and examples

3.2.6.1 Possible supported entry points to the Standalone Messaging

From the RCS user experience, the following three possible entry points to the Standalone Messaging may be supported:

1. Standalone Messaging screen/window
   There may be a dedicated "Standalone Messaging" application point of entry in the device menu. From this Standalone Messaging screen/window, a standalone message can be initiated or received using the relevant menu items and the device's supported keypad/keyboard. This application may also provide access to the user's message store for viewing and managing stored messages, e.g. message history.
2. Integrated messaging screen/window
   There may be a dedicated integrated messaging application point of entry in the device menu. From this integrated screen/window, a message can be initiated or received using the relevant menu items and the device's supported keypad/keyboard. This application may also provide access to the user's message store for viewing and managing stored messages, e.g. message history.
3. Address book window
   Using this entry point, a message may be initiated with any contact. The experience when interacting through this entry point is identical to that of the messaging screen/window.

NOTE: When displaying the messages exchanged, the time indication can be set according to the CPIM *DateTime* header and the *datetime* element in the delivery notifications as described in section 3.3.6.6 allowing correct ordering of the messages even if the device's clock is not set correctly.

3.3 1-to-1 Chat

3.3.1 Feature description

3.3.1.1 General

The Chat service enables users to exchange messages between two users instantly.

The following RCS 1-to-1 Chat features are described:

- Store and forward
  This feature requires a Messaging Server to store messages and notifications (delivery and display) when the destination user is not online and deliver them to the user when he comes online again (i.e. store and forward).

- Interworking of Chat to SMS/MMS
  This feature requires a Messaging Server to interwork the messages to and from SMS or MMS.
- Message revocation request of Chat messages
  This feature allows a client or originating Messaging Server to request for an undelivered Chat message to be revoked.
  The revocation process is not user driven but a technical enabler for clients or the Messaging Server on the originating network.
- Message revocation processing of Chat messages
  This feature requires a Messaging Server to process MessageRevoke requests and respond with a MessageRevokeResponse request based on the chat message delivery status.
- "Delivered" message notification
  This allows the sender of a message to be notified when their message has been delivered to the recipient.
- "Displayed" message notification
  This allows the sender of a message to be notified when their message has been displayed on one of the recipient’s devices. Note that this notification cannot certify that the recipient has actually read the message. It can only indicate that the message has been displayed on the recipient's terminal User Interface (UI).
- Delivery of notifications (delivered and displayed) outside a session
  It should be possible to deliver notifications independently of whether a 1-to-1 chat is established or not.
- IsComposing indications
  This allows a user in a chat conversation to see when another user is typing a new message/reaction.
- Local Black List
  The terminal/client may support a locally stored Black List to handle incoming chat requests. Users are allowed to qualify undesired incoming chat as spam. This prevents subsequent messages from those originators to be shown or even notified to the user. Also, this undesired traffic will not be acknowledged to have been read. The Black List behaviour applies not only to Chat but also to File Transfer.
- PNB
  The PNBs stored in the network and set by the RCS user contains the lists of URIs for contacts (or lists), that an RCS user has set for blocking purposes. The BPEF uses the PNB lists for chat incoming and outgoing traffic blocking.
- Local Conversation History
  The terminal/client supports a locally stored conversation.
- Support for the Common Message Store
  The Common Message Store may be used to synchronise chat messages between devices. It also allows the user to keep a back-up of important conversations in the network.
  In the device, alignment is expected between the local Conversation History and the synchronisation with the Common Message Store.
- User Alias (Display Name)
  A user defined display name may be sent when initiating a communication with another user.

- Multimedia during a chat conversation
  Multimedia exchange during a chat conversation is always provided via File Transfer as per section 3.5.

3.3.2 Interaction with other RCS features

3.3.2.1 Switching to Group Chat

A Group Chat can only be initiated from a user on a Service Provider which has deployed a Messaging Server. It is optional for a Service Provider to provide the Group Chat functionality. Therefore, from the terminal perspective, if the configuration parameter GROUP CHAT AUTH (see Table 75) is configured to disable Group Chat, the terminal should not allow the user to add additional parties to the chat or start a Group Chat.

A 1-to-1 Chat can be converted into a Group Chat by either of the two Users A and B by adding new users to it. User A and User B are given the option in their UI to add one or more chat partners to the conversation. A user may be limited to the contacts known by their devices to be RCS users. Otherwise the originating user's Messaging Server needs to be prepared to potentially interwork messages to non-RCS Users via SMS or MMS.

A real time check of contacts capabilities may be performed when initiating a Group Chat (section 3.3.6.3). A new Group Chat composing window is created in the initiating device, for example, User A’s device, and the result of this check is visible here.

When User A sends the first message a new Group Chat is opened between all the selected users, and User A and User B as described in section 3.3.6.3.

For User B a new Group Chat composing window is created in the user’s device.

3.3.2.2 File Transfer within 1-to-1 chat and interaction with the blacklist

NOTE: In the present section, the BPEF as described in section 2.15.1 may be provided by the Messaging Server.

During a 1-to-1 chat, either user is able to initiate a File Transfer from the chat composing window towards the other user. The File Transfer is established using a new SIP session and is carried in a new MSRP session which is different from the one used for the chat session.

If PNB is supported, the handling by the BPEF is same as in section 3.5.4.5. Note that in the case of File Transfer during chat, the sender of the file transfer needs to be checked again but against the FT blacklist this time ‘*rcs_pnb_ft_blockedusers*’.

On the device involved in the chat, the receiving user receives the File Transfer invitation inside the chat window with the sending user and is able to accept or decline it from that window. In a multidevice environment, the File Transfer invitation is also shown on the other devices of the user allowing them to accept or decline the invitation also from those devices.

If the user accepts the File Transfer, the terminal will either ask the user the location to store the file or use a default directory. Once received, the user can open the file from the chat composing window.

Please note that the spam/blacklist behaviour applies to File Transfer, and not only to Chat messages. If an invitation to receive a file is received from a blacklisted user, the client/device implementation should, from the UI point of view, not notify the user on receipt of a File Transfer invitation from a blacklisted sender. Instead it should log the event in the spam folder (e.g. “User A tried to send a file on TIME/DATE”).

See section 3.5 for more details on File Transfer.

3.3.3 High Level Requirements

The following list of high level requirements applies to 1-to-1 chat:

- Clients/devices:

3-3-1 "Delivered" message notification request and response
3-3-2 "Displayed" message notification request and response

NOTE: The client device should allow the user to enable or disable the displayed notifications request and response.

3-3-3 Delivery of notifications (delivered and displayed) outside a session
3-3-4 IsComposing indications
3-3-5 Procedures associated with the store and forward of both messages and notifications performed by the Messaging Server
3-3-6 Sending MessageRevoke requests

- Messaging Server: In addition to the requirements presented above.

3-3-7 Store and forward of both messages and notifications
Please note this is a function which is provided on the terminating Service Provider’s network, however, a Messaging Server may additionally provide originating store and forward to avoid dependencies with another Service Provider network’s implementations.
3-3-8 Interworking of Chat to SMS/MMS
3-3-9 Sending MessageRevoke requests
3-3-10 Handling of MessageRevoke requests

3.3.4 Technical Realisation

Two different technical realisations of 1-to-1 chat are available: OMA SIMPLE IM as described in [RCS-SIMPLEIM-ENDORS] or OMA CPM as described in [RCS-CPM-CONVFUNC-ENDORS]. The first sub-section describes the features that are common to both technical realisations, while the following two sub-sections describe what is unique to the individual technical realisations. The CHAT MESSAGING TECHNOLOGY configuration parameter defined in Table 75 determines the technical realisation used for 1-to-1 Chat.

3.3.4.1 Technical Realisation of 1-to-1 Chat features common to both OMA SIMPLE IM and OMA CPM

At a technical level, the Chat service implemented using OMA SIMPLE IM or OMA CPM relies on the following concepts:

- SIP procedures for the setup of sessions using MSRP for the message exchange;
- In the SDP of the SIP INVITE request and response, the *a=accept-types* attribute shall include only *message/cpim* and *application/im-iscomposing+xml*, i.e., “*a=accept-types:message/cpim application/im-iscomposing+xml*”.

- When a session is set up, messages are transported in the MSRP session. Each MSRP SEND request contains a request to receive an Instant Messaging Disposition Notification (IMDN) 'delivery' notification, and possibly a request to receive an IMDN 'display' notification. A client should, therefore, always include "positive-delivery" in the value for the CPIM/IMDN Disposition-Notification header field. That means that the value of the header field is either "positive-delivery" or "positive-delivery,display" depending on whether display notifications were requested. The value of "negative-delivery" is not used in RCS for 1-to-1 Chat.
  The receiving devices must generate an MSRP SEND request containing the IMDN status when the user message is delivered and if requested, another MSRP SEND request when the message is displayed.

NOTE: If there is not an already established MSRP session between sender and receiver, the Pager Mode (i.e. SIP MESSAGE) is used to transport IMDNs (delivery notification, display notifications)

- In normal circumstances between two users at most only a single session is active at a time. A client shall, therefore, not initiate a new Chat session towards a user with whom there is already an established Chat session.
- IMDN [RFC5438]: RCS relies on the support of IMDN as defined in [RFC5438] and [RFC5438Errata] to request and forward disposition notifications of all the exchanged messages (See also section C.2 for the errata mentioned in [RFC5438Errata]).
- In MSRP requests, the client shall set both the CPIM From and CPIM To headers to *sip:anonymous@anonymous.invalid* to prevent revealing the user's identity when transmitted over unprotected links. A client receiving a CPIM message in a one-to-one Chat should, therefore, ignore the identity indicated in the CPIM headers.
- The CPIM/IMDN wrapper shall be UTF-8 encoded to avoid any potential internationalisation issues.
- IMDN message identification for all messages (including those conveyed in the SIP INVITE and notifications delivered via SIP MESSAGE) as defined in [RFC5438].
- The originating Messaging Server shall always set the CPIM DateTime header in the chat messages and notifications it receives by overwriting the value provided by the client. A client receiving these requests should, therefore, rely on these headers rather than on locally available time information.
- Both the Originating and the Terminating function shall ensure that messages are received in correct order by the RCS client regardless if the messages are store and forwarded or not.
  - To achieve this, the terminating side shall wait for the delivery notification or 180 ringing response until a new message is sent if the message is carried in the INVITE.
  - As in the case for messages composed while offline (see section 2.7.1.1) when a message is carried in the INVITE request, the client shall wait for a provisional response before sending a new message.
- For network optimisation purposes, the aggregation of IMDNs as specified in [RFC5438] may be supported for network initiated IMDNs:

  - Within the Service Provider's own network, the aggregation of IMDN may be supported (per local policy).
  - For inter-Service Provider interoperability, the individual IMDN will always be sent to the target network, where the aggregation of IMDN is up to the target network (per local policy). That is, if the aggregated IMDNs received by the Messaging Server contain IMDNs that need to be sent to another network, the Messaging Server will repackage the aggregated IMDNs accordingly before sending them to the Chat message sender on the other network.
  - If the aggregated IMDNs received by the Messaging Server contain both in-network and inter-Service Provider Chat message senders, the Messaging Server will repackage the aggregated IMDNs according to in-network Chat message senders and inter-Service Provider Chat Message senders.

- Auto-acceptance of store and forward Messaging Server PUSH of stored notifications.
- Store and forward Messaging Server PUSH of stored messages.
- Chat inactivity timeout: When a device or the network detects that there was no activity in a chat for IM SESSION TIMER, a configurable period of time (see Table 75), it will close the established Chat session.
- When reopening an older chat on the device, that contains messages for which a "display" notification should be sent, these notifications shall be sent as follows:

  - If there is no session established with the sender, the device will send the notifications outside a session (since there is no current session to send them to) using SIP MESSAGE;
  - If there is an active session but that session is with a device of the sender other than the one that was used to send the message to which this notification relates, the Messaging Server will ensure that these notifications are delivered outside of that session;

- The "IsComposing" notification is generated and processed according to the rules and procedure of [RCS-SIMPLEIM-ENDORS] and [RCS-CPM-CONVFUNC-ENDORS]. Consequently, the 'IsComposing' notification is not sent with CPIM headers, and as such a delivery and/or displayed notification cannot be requested.
- The transfer of files while a Chat session is taking place shall at protocol level be performed in a separate session. From the user experience perspective, they should be able to transfer files whilst in chat. Messages over a maximum size (MAX SIZE IM in section A.1.4.3) should be transferred using File Transfer. All multimedia content shall be transferred using File Transfer.

3.3.4.1.1 Client Side Spam/Black List Handling

When receiving a message from a sender included in the Black List (i.e. a spam sender) the receiving client's/device's implementation shall:

- Terminate the transaction with a 486 BUSY HERE sent back to the sender.
- The receiver will still issue a delivery notification with status "delivered" which will be sent back to the sender.

- From the UI point of view, the receiver should not be notified on the reception of a message from a blacklisted sender and the message should be copied to the spam filter.

3.3.4.1.2 Personal Network Blacklists handling

NOTE: In the present section, the BPEF as described in section 2.15.1 may be provided by the Messaging Server itself.

When supported and enabled, the PNB described in section 2.15 are applied by the BPEF at both origination and termination of 1-to-1 chat invitation requests.

The following *resource-lists* from Shared XDMS are checked by the BPEF by comparing the URI values used in the request and in the list:

- On origination:
  a) Upon initiation of the 1-to-1 chat, the BPEF of the originator checks the '*rcs_pnb_outchat_blockedusers*' list to verify that the recipient is not among the blocked users for this request by comparing URIs contained in the list with the URI value of the Request URI of the SIP request.
  b) If found, the BPEF shall reject the chat with a *403 Forbidden* with a warning header set to "122 Function not allowed" towards the user without forwarding the SIP INVITE to the recipient's network.

- On termination:
  a) The BPEF checks the *'rcs_pnb_chat_blockedusers*' list, to verify if the originator of the chat is among the blacklisted users by comparing the URIs contained in the list with the URI values of the P-Asserted-Identity header field of the SIP request.
  b) If the sender is among the blacklisted users, the BPEF returns a *403 Forbidden* with a warning header set to "122 Function not allowed" to the originator's network, without forwarding the SIP INVITE to the recipient.
  c) If the Common Message Store feature is supported, it stores the Session History folder data as defined in [RCS-CPM-MSGSTOR-ENDORS] for the blocked chat invite event.

3.3.4.1.3 Chat abnormal interruption

If a device in a chat suffers an abnormal termination of the Chat session, for example, loss of coverage, the "Send" button may be disabled. If the device determines that a message could not be sent (e.g. failed response or received no response), it shall inform the user that the chat message was not sent. If the TCP connection is lost, the client should re-send it in a new chat session once re-registered.

NOTE1: If the Messaging Servers involved in the chat have implemented store and forward functionality, then the Messaging Servers shall be responsible for storing any messages received while a chat has been abnormally interrupted.

NOTE2: In temporary interruption cases, for example, a device was out of network coverage but is now again within network coverage, the chat can be continued from the same conversation window. In this case a new session has to be established with a SIP INVITE request.

3.3.4.1.4 Store and Forward Mode

The store and forward functionality in the network is optional and it is up to each Service Provider to decide whether to deploy it.

The store and forward functionality requires a Messaging Server. There are three possible scenarios to fulfil the requirement for store and forward functionality:

1. Sender and receiver are on networks with a Messaging Server supporting store and forward: In this case, the receiver's side Messaging Server has the responsibility to store and forward IMs which are not delivered. The sender's side Messaging Server has the responsibility of storing the delivered/displayed notifications if the sender is offline.
2. Only the sender is on a network with a Messaging Server supporting store and forward: The sender's side Messaging Server has the responsibility to store and forward Chat messages and/or delivered/displayed notifications if immediate delivery was not possible. As it is in the sender's network, the Messaging Server will not have information on when the receiver is online; therefore a retry mechanism is used. Note that it is the Service Provider's decision whether they provide store and forward for chat messages on behalf of the receiver who is in a different network that does not support store and forward.
3. Only the receiver is on a network with a Messaging Server supporting store and forward: The receiver's side Messaging Server has the responsibility to store and forward Chat messages and/or delivered/displayed notifications if they cannot be delivered. As it is at the receiver's side, that Messaging Server will not have information on when the sender is online. Therefore, a retry mechanism is used to store and forward notifications that could not be delivered right away.

NOTE: Whether a Service Provider provides store and forward for delivered/ displayed notifications on behalf of the sender who is in a different network that does not support store and forward is optional.

With the introduction of application messages making use of the chat session (see section 3.5.4.8 and 3.10.4), the store and forward functionality for chat will have to deal with those content types. That shall be done as follows:

- When accepting a Chat session on behalf of a user, a Messaging Server shall indicate support for the application message content types that it supports (e.g. in the *a=accept-wrapped-types* SDP attribute that it provides in the SIP 200 OK Response.
- Storage of such content shall be as any other message content.
- When a client comes online, forwarding shall be as follows:
    - When included as a body of a SIP INVITE request, for non-text content a dedicated Accept-Contact header field shall be added to the INVITE request

carrying the IARI defined for the service corresponding to the included content type in section 2.6.1.1.2 with the *require* and *explicit* parameters.

- If delivery fails with a SIP 480 response, the Messaging Server shall store the original message and forward it later (again including an *Accept-Contact* header field as for the initial forward) when another device of the user comes online.
- The message shall also remain stored for later delivery to another device if when the application message needs to be forwarded in a session that is already established and the *a=accept-wrapped-types* attribute provided by the client that accepted the session didn't include support for the corresponding MIME type.

The Messaging Server stores undelivered messages for a period that is determined by local server policy. If at the end of this period the messages have not been delivered, the Messaging Server discards them. This applies to notifications as well as messages.

A dedicated configuration setting (IM CAP ALWAYS ON, see Table 75 in Annex A for further reference) is used to configure the client to allow sending messages to offline users.

NOTE: The procedure for Messaging Server to store the chat message when the participant is temporarily unavailable is described in [RCS-SIMPLEIM-ENDORS] or [RCS-CPM-CONVFUNC-ENDORS] based on the CHAT MESSAGING TECHNOLOGY configuration parameter defined in Table 75.

3.3.4.1.5 Delivering stored disposition notifications

To be able to deliver delivered/displayed notifications via store and forward to a sender's device that has come online again, without disrupting the user experience, the Messaging Server supporting the store and forward functionality shall initiate a special session for the purpose of delivering these notifications. This special session shall be automatically accepted by the device. It is recognized by the device by means of the well-known username part of the URI (*rcse-standfw@<domain>*) uniquely identifying the store and forward service identity that is provided in the *P-Asserted-Identity* header field. Optionally an operator can disable the delivering of the stored notifications when the RCS user is roaming in a foreign network.

NOTE1: The Messaging Server may also use Pager Mode messaging to deliver stored delivery and displayed notifications.

The Messaging Server supporting the store and forward functionality is required to send the delivered/displayed notifications when the first device of the user comes online.

NOTE2: The procedure for Messaging Server to deliver the stored chat messages and associated disposition notifications are described in [RCS-SIMPLEIM-ENDORS] or [RCS-CPM-CONVFUNC-ENDORS] based on the CHAT MESSAGING TECHNOLOGY configuration parameter defined in Table 75.

3.3.4.1.6 Interworking towards SMS/MMS

The functionality for interworking of the chat service to SMS/MMS is optional and it is the decision of each Service Provider whether to deploy it. This deployment involves:

- The Messaging Server described in [RCS-SIMPLEIM-ENDORS] or [RCS-CPM-CONVFUNC-ENDORS].

- The ISF described in [RCS-CPM-IW-ENDORS] which is responsible for selecting the appropriate interworking function for a new session.
- The IWF for SMS and MMS described in respectively [RCS-3GPP-SMSIW-ENDORS] and [RCS-CPM-IW-ENDORS] which are responsible for doing the actual interworking (that is the protocol conversions) between RCS based chat and SMS or MMS.

Based on service-level agreements (SLAs), interworking of chat may occur on the originating side or the terminating side, the same circumstances as for interworking of messages with SMS/MMS described in section 3.2.4.6. In brief, the interworking is initiated by the Messaging Server either based on local information it may have about the recipient, or when it receives one of the following error responses on the INVITE request that indicate that the recipient is not an RCS contact:

- 404 Not Found;
- 405 Method Not Allowed;
- 410 Gone;
- 414 Request URI Too Long;
- 415 Unsupported Media Type;
- 416 Unsupported URI Scheme;
- 488 Not Acceptable Here;
- 606 Not Acceptable.

3.3.4.1.6.1 Interworking at Originating Side

When a Chat session invitation needs to be interworked on the originating side, the CPM Participating Function will route the invitation to the ISF, which will select either SMS or MMS interworking based on applicable service provider policy. The ISF will then route the message to the selected IWF, which will either accept the chat invitation automatically on behalf of the SMS/MMS user, or will convert the chat invitation to an SMS/MMS invitation message and deliver it to the terminating network using the appropriate SMS/MMS NNI. The response to the chat invitation from the SMS/MMS user must be received through the same SMS/MMS interface to associate correctly the response with the earlier invitation. The SMS/MMS response (either accept or decline) to the invitation is converted to the appropriate SIP response and conveyed back to the RCS user.

3.3.4.1.6.2 Interworking at Terminating Side

When a Chat session invitation needs to be interworked on the terminating side, the invitation will be first routed to the terminating network as described in previous sub-sections, and then the same procedures as for interworking of chat invitations on the originating side will apply.

3.3.4.1.7 Multidevice handling

Multidevice handling occurs when a user has more than one device (e.g., PC and mobile).

When a new 1-to-1 chat is initiated and a message is sent from User A to a User B with User B having multiple devices registered at the same time, the network or Messaging Server forks the Chat session invitation to the different devices. Forking on the Messaging Server is further elaborated in section 3.3.4.1.7.1.

NOTE: It is assumed that the originating user uses one device per session.

Each of User B's devices that receive the session invitation with a message in the INVITE generates a SIP MESSAGE request to carry the delivered IMDN. In a multidevice scenario, if a sender receives more than one IMDN for a sent message, it shall discard all copies except the first one it receives.

User B is able to respond to the chat from any of their devices. When they answer and send a message from one of the devices, that device (B1) becomes the only active device for User B and all the Chat sessions towards the other devices are terminated.

Once the user has answered the chat from device B1, all the subsequent messages sent to User B are received only by the active device B1 using the already established Chat session.

Device switching:

1. If User B closes the Chat session from the active device (either by closing the chat conversation from the chat window or due to an abnormal termination), any new messages sent by User A through the chat will make the Messaging Server establish the chat again using one Chat session per connected device of User B and send the message to them all.
2. If User B changes from one device B1 to another B2 by sending a new message to the chat from the new device B2, B2 will send a new INVITE request that will go to User A's device. When User A's device detects a new INVITE request from User B which already has an established session with User A's device it shall end that session and accept the new one. All subsequent messages are received only by device B2. Device B2 must then store the received messages and display them appropriately.

3.3.4.1.7.1 Forking on the Messaging Server

Forking to registered online devices in case that there is no message in the incoming INVITE request shall be achieved by using the forking capability at the Messaging Server. In case there is a message in the incoming INVITE request, forking to registered online devices may be achieved by using the forking capability at the Messaging Server according to Service Provider policy. This capability shall be implemented on the terminating Participating Function.

As described in section 3.3.4.1.7, in case of an incoming INVITE request, User B is able to respond to the chat from any of their devices. When User B responds from one of the devices, that device becomes the only "active" device (i.e. 'is composing' notifications or messages originated from that device) for that user and, consequently, the terminating Participating Function shall tear down all other chat sessions towards the other devices under the same chat session identity by sending a SIP BYE request including a Reason header field with the protocol set to SIP and the protocol-cause set to 200 along with an optional protocol-text (e.g. SIP;cause=200;text="Call completed elsewhere"). A client may use this information to update its representation in the UI.

When a device belonging to User B registers in IMS and provided there is no other active device, the terminating Participating Function shall send an INVITE request for that chat session to the newly registered device.

3.3.4.1.8 Emoticons

Selected emoticons are displayed graphically but sent and received as text. The list of supported emoticons is defined in [RCS-SIMPLEIM-ENDORS] Appendix N.

3.3.4.1.9 Chat message size limitations

The maximum size is controlled through the MAX SIZE IM configuration parameter defined in Table 75. Messages that are larger than the maximum size indicated in the MAX SIZE IM configuration parameter can be delivered either using File Transfer or a Large Message Mode standalone message.

3.3.4.1.10 Message Revocation

Message revocation is a feature that allows a client or Messaging Server to request for a chat message to be revoked by the recipient's Messaging Server. The recipient's Messaging Server processes MessageRevoke requests and responds with a Message Revoke Response request based on the chat message delivery status.

3.3.4.1.10.1 Generating Chat Message Revoke Requests

The MessageRevoke request is generated by either the client or the Messaging Server of the message sender, depending on the Service Provider policy. The MessageRevoke request is carried in the body of a SIP MESSAGE request that includes the same imdn.message-ID value of the chat message that is intended to be revoked (as described in section 3.3.4.1.10.4).

MessageRevoke requests shall be generated only towards networks where their Messaging Server can handle them as described in section 3.3.4.1.10.2 and are not meant to reach other clients. MessageRevoke requests shall not be generated in case the delivery notification pertaining to the original message has been received.

NOTE: Given that a revoke may be sent only if support has been indicated by the terminating network, it cannot be initiated when the INVITE transaction is still pending.

3.3.4.1.10.1.1 Message Revoke Requests by the Client

When message revocation is enabled (CHAT REVOKE TIMER, see section A.1.4.3), the client can generate MessageRevoke requests once the timer is expired. In order for the MessageRevoke requests to be transmitted, the client shall have data connectivity.

MessageRevoke requests shall be generated only if support for MessageRevoke requests has been indicated in the Contact header of the SIP INVITE request or in the response to the SIP INVITE request of the session to which the message intended to be revoked belongs. Specifically, this indication is in the form of a feature tag in the Contact header of the SIP INVITE request or response that is defined in section 3.3.4.1.10.3. For session initiation scenarios that result in SIP *486 Busy Here* responses from the terminating Messaging Server, a feature tag cannot be included by the terminating network since according to [RFC3261] a Contact header is not present in such responses. However, depending on Service Provider policy a MessageRevoke request may be generated (for the case where a first message is included in the SIP INVITE request). In that case, this MessageRevoke request might be blocked on the NNI if not supported by the terminating network (see section 3.3.5.3).

When a message is to be revoked, the client shall include an Accept-Contact header field with either the CPM ICSI for Session Mode Messaging, or the OMA SIMPLE IM feature tag (depending on the value of the CHAT MESSAGING TECHNOLOGY configuration parameter defined in Table 75), as is already the case for IMDNs carried in SIP MESSAGE requests, and shall add a dedicated Accept-Contact header field carrying the Message Revoke feature tag defined in section 3.3.4.1.10.3 along with the *require* and *explicit* parameters. The client shall also include the message revocation content-type including the value of the imdn.message-ID of the original message that is requested to be revoked, as described in section 3.3.4.1.10.4. The Request-URI of the MessageRevoke request shall be set to the address of the target contact of the message that is requested to be revoked. The body of the MessageRevoke request, as described in section 3.3.4.1.10.4, shall have:

- The <Message-ID> element set to the value of the imdn.message-ID of the original message that is requested to be revoked,
- The <From> element set to the URI of the sender of the message,
- The <To> element set to the URI of the recipient of the message.

3.3.4.1.10.1.2 Message Revoke Requests by the Messaging Server

Similarly to section 3.3.4.1.10.1.1, MessageRevoke requests shall be generated only if support for MessageRevoke requests has been indicated in the Contact header of the SIP INVITE request or in the response to the SIP INVITE request of the session to which the message intended to be revoked belongs. Specifically, this indication is in the form of a feature tag in the Contact header of the SIP INVITE request or response. For session initiation scenarios that result in SIP *486 Busy Here* responses from the terminating Messaging Server, a feature tag cannot be included by the terminating network. However, depending on Service Provider Policy, a Message Revoke request may be generated (for the case where a first message is included in the SIP INVITE request). In that case, this MessageRevoke request might be blocked on the NNI if not supported by the terminating network (see section 3.3.5.3).

The format of the MessageRevoke request generated by the Messaging Server is the same as described for the MessageRevoke requests by the Client in section 3.3.4.1.10.1.1.

3.3.4.1.10.2 Handling MessageRevoke Requests

For a network, handling of MessageRevoke requests goes along with having the functionality to generate MessageRevoke requests (either by the client or by the Messaging Server) and vice versa.

The MessageRevokeResponse request shall indicate the result of the MessageRevoke request that can be either successful or failed. Similarly to the MessageRevoke request, the MessageRevokeResponse request include an Accept-Contact header field with either the CPM ICSI for Session Mode Messaging, or the OMA SIMPLE IM feature tag (depending on the value of the CHAT MESSAGING TECHNOLOGY configuration parameter defined in Table 75), as it is already the case for IMDNs carried in SIP MESSAGE requests, and shall add a dedicated Accept-Contact header field carrying the Message Revoke feature tag defined in section 3.3.4.1.10.3 without the *require* and *explicit* parameters. The Messaging Server handling the MessageRevoke request shall also include the message revocation content-type including the value of the imdn.message-ID of the original message that was requested to be revoked, and the revoke result parameter as described in section 3.3.4.1.10.4. The MessageRevokeResponse request shall include the device identifier (GRUU or sip.instance feature tag) as specified in section 2.11.2 based on the device identifier included upon chat session establishment. The Request-URI of the MessageRevokeResponse request shall be set to the address of the contact that sent the message that is requested to be revoked.

The MessageRevokeResponse request shall be indicated as successful when the message to be revoked is removed from the deferred storage and will therefore not be delivered to the client.

The MessageRevokeResponse request shall be indicated as failed when any of the following conditions is met:

- Interworking towards SMS/MMS has occurred at originating or terminating side
- A successful delivery notification for which the MessageRevoke request has been generated has been received by the originating or terminating Messaging Server;
- Message revocation is not performed successfully by the terminating Messaging Server (e.g., due to Messaging Server failures);
- The message that the intended MessageRevoke request has been generated for is stored at the terminating side in the Common Message Store.

MessageRevoke requests shall never be forwarded to the client and shall be processed right after being received by the Messaging Server.

3.3.4.1.10.3 Message Revoke feature tag

RCS defines a Message Revoke feature tag to indicate support of the message revocation feature. The RCS Client and originating Messaging Server shall make use of the message revocation feature only when the terminating Messaging Server has indicated its support through the Message Revoke feature tag. It can be used to indicate support for revoking of any message identified with a CPIM Message-ID. However, this release of RCS only allows it for chat messages.

The feature tag is set in the Contact header of the SIP INVITE request or response used to set up a 1-to-1 chat session and it is always attached by the Messaging Server that supports Message revocation feature. The client shall only include this feature tag in the MessageRevoke request.

The feature tag is defined as *+g.gsma.rcs.msgrevoke*.

3.3.4.1.10.4 Message Revoke content-type

The Message Revoke XML schema is defined as shown on Table 30.

The associated MIME content type is *application/vnd.gsma.rcsrevoke+xml*.

This content type used in both the MessageRevoke request and in the MessageRevokeResponse request.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:gsma:params:xml:ns:rcs:rcs:rcsrevoke"
        xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns="urn:gsma:params:xml:ns:rcs:rcs:rcsrevoke"
        elementFormDefault="qualified"
        attributeFormDefault="unqualified">
    <xs:element name="imRevoke">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="Message-ID" >
                    <xs:simpleType>
                        <xs:restriction base="xs:token"/>
                    </xs:simpleType>
                </xs:element>
                <xs:element name="result" minOccurs="0" maxOccurs="1">
                    <xs:simpleType>
                        <xs:restriction base="xs:string">
                            <xs:enumeration value="success"/>
                            <xs:enumeration value="failure"/>
                        </xs:restriction>
                    </xs:simpleType>
                </xs:element>
                <xs:element name="From">
                    <xs:simpleType>
                        <xs:restriction base="xs:anyURI"/>
                    </xs:simpleType>
                </xs:element>
                <xs:element name="To">
                    <xs:simpleType>
                        <xs:restriction base="xs:anyURI"/>
                    </xs:simpleType>
                </xs:element>
                <xs:any namespace="##other" processContents="lax" minOccurs="0"
                maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>
```

Table 30: RCS Revoke and RevokeResponse message body schema

The following is an example of the body of a SIP MESSAGE requesting that a specific chat message be revoked. In order to know whether the revoke was successful or not, the MessageRevoke request sender checks the result field in incoming MessageRevokeResponse requests.

Example of a MessageRevoke request:

```
Content-type: application/vnd.gsma.rcsrevoke+xml
Content-length: ...

<?xml version="1.0" encoding="UTF-8"?>
<imRevoke xmlns="urn:gsma:params:xml:ns:rcs:rcs:rcsrevoke">
<Message-ID>23499fuq34fu</Message-ID>
<From>tel:+1234578901</From>
<To>tel:+1234578902</To>
</imRevoke>
```

NOTE: The Message-ID, "23499fuq34fu", in the XML body refers to the CPIM Message-ID of the message to be revoked.

Example of a MessageRevokeResponse request where the revoke succeeded. If it had failed, the value of result would be “failed”:

```
Content-type: application/vnd.gsma.rcsrevoke+xml
Content-length: ...

<?xml version="1.0" encoding="UTF-8"?>
<imRevoke xmlns="urn:gsma:params:xml:ns:rcs:rcs:rcsrevoke">
<Message-ID>23499fuq34fu</Message-ID>
<result>success</result>
<From>tel:+1234578901</From>
<To>tel:+1234578902</To>
</imRevoke>
```

NOTE: The Message-ID, "23499fuq34fu", in the XML body refers to the CPIM Message-ID of the message that was revoked.

3.3.4.1.11 1-to-1 Chat Service Identification

The RCS client shall populate the P-Preferred-Service header field in all CPM and SIMPLE IM requests with the CPM Feature tag defined for the service, as described in [RCS-CPM-CONVFUNC-ENDORS]. The S-CSCF or AS that performs the service assertion in the originating network shall add the P-Asserted-Service header field set to the value of the asserted CPM service ICSI (i.e. “*urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session*” for CPM chat, or “*urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred*” for deferred delivery done as part of the Store and forward realization) and remove the P-Preferred-Service header field before further routing the request.

NOTE: During a transition period towards full compliance, the network support for asserting the service is recommended but not mandatory.

A receiving network element and RCS client should ignore any SIP header fields that they do not understand (e.g. P-Preferred-Service, or P-Asserted-Service header fields).

3.3.4.2 Technical Realisation of 1-to-1 Chat features when using OMA SIMPLE IM

At the technical level, the 1-to-1 Chat service implemented using OMA SIMPLE IM extends the concepts described in section 3.3.4.1 with the following concepts:

- For OMA SIMPLE IM, first message is always included in a CPIM/IMDN wrapper carried in the SIP INVITE request. A client should always include “positive-delivery” in the value for the Disposition-Notification header field in that message. That means that the value of the header field is either “positive-delivery” or “positive-delivery,display” depending on whether display notifications were requested. The value of “negative-delivery” is not used in RCS for 1-to-1 Chat. SIP INVITE requests for a one-to-one session that carry a message in CPIM/IMDN wrapper shall be rejected by the server unless they carry a Disposition-Notification header that at least includes “positive-delivery”.

- If auto-accept is not used, then the devices each send a SIP 180 response toward A.
- The received Chat session invitation contains an IMDN requesting 'delivery' notification. So each receiving device sends back a SIP MESSAGE request containing the IMDN indicating successful delivery of the original message sent by A.
- The receiving clients each send a 486 BUSY HERE response to the outstanding INVITE when a new INVITE arrives from the same user so that there is not more than one outstanding INVITE from one user. The IMDN for 'delivery' notification is requested and sent in the same way as described above.
- No support for exchanging multimedia content within a chat. Therefore, in the SDP of the SIP INVITE request and response, the *a=accept-wrapped-types* attribute shall only include text/plain and message/imdn+xml. If File Transfer using HTTP is supported (see section 3.5.4.8) then the *a=accept-wrapped-types* attribute shall also include *application/vnd.gsma.rcs-ft-http+xml*. If Geolocation PUSH is supported (see section 3.10.4.1.2), then the *a=accept-wrapped-types* attribute shall also include *application/vnd.gsma.rcspushlocation+xml*. To transfer multimedia content during a chat, File Transfer is used.
- When one of User B's devices detects user activity relevant to the consumption of the message contained in the invitation (e.g. click on a pop-up to go to the Chat window) a 1-to-1 chat session is established according to the following possible criteria:

  a) The respective client returns a 200 OK response, signalling the initiation of the remaining procedures to establish the chat when User B reacts to the notification by opening the chat window. This is the default criteria for RCS and, consequently, all the diagrams shown in this document reflect this behaviour.
  b) The 200 OK response is sent when User B starts to type a message, or
  c) The 200 OK response is sent when User B sends a message. Please note that in this case User B’s message will not generate an invite but is buffered in the client until the MSRP session is successfully established.
  d) The 200 OK response is sent immediately since the devices receiving the invitation are configured to auto-accept the session invitations (IM SESSION AUTO ACCEPT configuration parameter defined in Table 75).

  Please note that:

  - The behaviour for criteria a), b) and c) is configured via the IM SESSION START parameter as defined in Table 75. The behaviour for criteria d) is configured via the IM SESSION AUTO ACCEPT configuration parameter defined in Table 75.
  - For a), b) and c), the 200 OK is sent if the chat invitation has not expired. Otherwise, User B's message shall be sent in a new invitation (from User B to User A).

If the Chat session invitation from User A contained an IMDN *Disposition-Notification* header requesting a 'display' notification and if the privacy settings allow it, the device User B is using shall generate an MSRP SEND request toward User A that contains the IMDN 'display' status for the message received from User A.
It may be the case that multiple Chat sessions from User A are pending on User B's side, that is the last received Chat session is established and the other pending sessions are answered with a 486 BUSY HERE response. In such cases, if the Chat session invitations from User A contained a IMDN Disposition-Notification header requesting a 'display' notification, the device of User B that accepted the SIP INVITE generates an MSRP SEND request toward User A that contains the IMDN 'display' status for each message received from User A.

NOTE: The statement in section 3.3.4.1 that the CPIM/IMDN wrapper shall be *UTF-8* encoded to avoid any potential internationalization issues also applies to the IMDN requested in the SIP INVITE request.

- A Messaging Server supporting store and forward behaves as a back-to-back user agent handling the SIP INVITE requests that are used to establish the chat session. While doing this it may have to return a different response to the INVITE request on the originating leg than the one it received on the INVITE request on the terminating leg. The mappings shown in Table 31 will be applied:

Response received on terminating leg	Response sent on originating leg	Store the message
480 Temporarily unavailable	200 OK	Y
408 Request Timeout	486 Busy Here	Y
487 Request Terminated	486 Busy Here	Y
500 Server Internal Error	486 Busy Here	Y
503 Service Unavailable	486 Busy Here	Y
504 Server Timeout	486 Busy Here	Y
600 Busy Everywhere	486 Busy Here	Y
603 Decline	486 Busy Here	Y
Any other response (including 404 Not Found and 200 OK)	Received response (that is no mapping is done)	N

Table 31: Mapping of received Error Responses by the Messaging Server

- To reduce the complexity at protocol level and avoid potential TCP switchover(s), it is recommended to limit the maximum size of a chat message (see section 3.3.4.1.9) to avoid the SIP INVITE request to be longer than the path MTU (e.g., 1300 bytes) and, consequently, trigger the TCP switchover. The maximum size controlled through the MAX SIZE IM configuration parameter defined in Table 75 applies to both the first message in the INVITE and to messages sent via MSRP. If the user attempts to send a first or subsequent chat message larger than this limit (counting the size of the CPIM body only, that is CPIM headers are not included in the count), then the user shall be notified that the message is too large.
- In the first message in the INVITE, the client shall set both the CPIM From and CPIM To headers to *sip:anonymous@anonymous.invalid* to prevent revealing the user’s

identity when transmitted over unprotected links. A client receiving a CPIM message in a one-to-one Chat should therefore ignore the identity indicated in the CPIM headers.

3.3.4.2.1 Clarifications on Chat race conditions

- Two simultaneous invites. Though unlikely, it may be possible that two users decide to invite each other simultaneously for a chat. In this situation the behaviour of the clients should be the following:
  - User A sends an invite to User B for Chat
  - Before a final response for that invite is received, User A receives an invite from User B for Chat
  - User A will send a 486 BUSY HERE response to User B. In addition to this, User A will send the correspondent delivery and read notification using SIP MESSAGE.
  - From the UX point of view, the message sent by B will be displayed as received.
  - User B will behave as user A, potentially resulting in both session invitations being turned down with a SIP 486 BUSY HERE response. Users will have to retry session setup until successful.
- New invite sent after a previous invite has been accepted. Though unlikely, the following scenario can take place:
  - User A sends an invite for chat to User B,
  - User B accepts the chat a 200 OK response is sent back to User A,
  - In parallel and before receiving the 200 OK response, User A sends a new invite with a new message.
- To resolve the race condition:
  - When User B receives the new invitation, it should terminate the current MSRP session (if established) by sending a SIP BYE.
  - Once the initial session is terminated, a new 200 OK response should be issued which will trigger the establishment of a new MSRP session.

For additional clarification, explanatory diagrams have been included in Annex B, sections B.1.9 and B.1.10.

3.3.4.3 Technical Realisation of 1-to-1 Chat features when using OMA CPM

At a technical level the Chat service implemented using OMA CPM extends the concepts described in section 3.3.4.1 with the following concepts:

- If auto-accept is not used, then the devices send a SIP 180 response toward A.
- When users are allowed to have multiple devices and those devices are configured to auto-accept (IM SESSION AUTO ACCEPT set to 1, as defined in section A.1.4.3), the Messaging Server is required to be able to fork the incoming 1-to-1 Chat session request to each of the receiving user's devices to set up an MSRP session with each of them.

- The receiving clients (or their Participating Function on their behalf) each send a 486 BUSY HERE response to the outstanding INVITE request when a new INVITE request arrives from the same user so there is not more than one outstanding INVITE request from one user.
- Multimedia content within a Chat session is not permitted. Therefore, in the SDP of the SIP INVITE request and response, the *a=accept-wrapped-types* attribute shall only include text/plain and message/imdn+xml and if File Transfer using HTTP or Geolocation PUSH is supported (see sections 3.5.4.8 and 3.10.4.1.2) *application/vnd.gsma.rcs-ft-http+xml* and *application/vnd.gsma.rcspushlocation+xml* respectively, e.g., *a=accept-wrapped-types:text/plain message/imdn+xml*. To transfer multimedia content during a chat, File Transfer is used.
- When one of User B's devices detects user activity relevant to the consumption of Chat session invitation (e.g. click on a pop-up to go to the Chat window) a 1-to-1 chat session is established according to the following possible criteria:

  a) The respective client returns a 200 OK response, signalling the initiation of the remaining procedures to establish the chat when User B reacts to the notification by opening the chat window. This is the default criteria for RCS and, consequently, all the diagrams shown in this document reflect this behaviour.
  b) The 200 OK response is sent when User B starts to type a message, or
  c) The 200 OK response is sent when User B sends a message. User B’s message is buffered in the client until the MSRP session is successfully established.
  d) The 200 OK response is sent immediately if the devices receiving the invitation are configured to auto-accept[28] the session invitations (IM SESSION AUTO ACCEPT configuration parameter defined in Table 75).

  Please note that the behaviour for criteria a), b) and c) is configured via the IM SESSION START parameter as defined in Table 75. The behaviour for criteria d) is configured via the IM SESSION AUTO ACCEPT configuration parameter defined in Table 75.

NOTE: Unlike the realisation with OMA SIMPLE IM described in section 3.3.4.2, the realisation of the 1-to-1 Chat service based on OMA CPM does not allow to carry the first Chat message in the SIP INVITE request as per [RCS-CPM-CONVFUNC-ENDORS].

3.3.5 NNI and IOT considerations

3.3.5.1 Chat session interworking when one side carries a message in the INVITE request

Interworking from a Chat session with a chat message in the INVITE request to a Chat session where the INVITE request does not carry any chat message requires that the Messaging Server (or a separate network entity) performing the interworking store the

[28] Note that the Service Provider multidevice policy has to be consistent with Chat auto-acceptance policy.

message in the INVITE until the Chat session without first message in INVITE is set up. If multiple Chat session INVITEs with chat messages arrive before the Chat session on the other side is set up, multiple chat messages are stored, however it is recommended that the Messaging Server automatically accept the session on behalf of a user in a network not supporting first message in the INVITE request. If no Chat session is set up on the other side, the chat messages are kept and delivery is attempted at a later time in the same way as already specified in section 3.3.4.1.4 when chat messages are stored on the originating side.

Interworking from a Chat session without first message in INVITE to a Chat session with a message in the INVITE requires that the Messaging Server accept the Chat session without any message on behalf of the recipient user and once the first chat message is received via MSRP, initiate an INVITE towards the recipient, including the first chat message as a CPIM body in the INVITE. Providing the recipient, or the recipient’s Messaging Server on behalf of the recipient, does not set up a session, the Messaging Server performing the interworking continues to generate INVITEs towards the recipient for each new chat message received.

See the flows in Annex B for more information.

3.3.5.2 SIMPLE IM session and CPM session interworking of feature tags

The mapping of the appropriate SIMPLE IM session feature tags is done as per Appendix G in [RCS-CPM-CONVFUNC-ENDORS] when it is determined that the remote network requires such interworking. Also once a session is set up with the recipient, the Messaging Server or a separate network entity performing the interworking ensures that messages exchanged via MSRP are sent end to end.

See the flows in Annex B for more information.

3.3.5.3 Interworking between a service provider that supports “Message Revocation” and a service provider that does not support

To prevent the MessageRevoke request being leaked to the service provider’s side that does not support message revocation, leading to unwanted user experience and possible undesired charging implication, the service provider that implements the message revocation shall make sure that the MessageRevoke requests are only sent to the service providers that also implement the message revocation when interworking.

3.3.6 Implementation guidelines and examples

Please note that where the specification describes the user interface, it should be taken as guidance.

3.3.6.1 General

End to end flows for store and forward 1-to-1 chat with notifications can be found in Annex B.

The following sections show the relevant chat message flows and reference user experience. Please note that the following assumptions have been made:

- For simplicity, the internal mobile network interactions are omitted in the diagrams shown in the following sections.

- Each Service Provider may deploy a Messaging Server (that is the use of a Messaging Server is optional in RCS deployments), to manage all messages from its customers.
- Prior to the chat, the user will have accessed their address book or Chat application to start the communication. As described previously, while these actions are performed an OPTIONS or Presence request is sent to verify the available capabilities. In the following diagrams it is assumed that this exchange (OPTIONS/Presence request and response) has already taken place, and therefore, both ends are aware of the capabilities and the available RCS services of the other side. If that is not the case, the OPTIONS (or Presence) request should be sent at the same time the chat is being set up.

Service Provider support of the store and forward functionality is optional in RCS. To allow a Service Provider to provide store and forward functionality to its customers even in cases where the Chat session is established towards a user of a Service Provider that does not support store and forward, the messages can optionally be stored and forwarded from the sender's Messaging Server, based on operator's policy.

3.3.6.2 Entry points to the chat service

From the UX perspective there are three possible entry points to this service:

1. Address book/Call-log: Chat can be initiated to any RCS contact with Chat capability as described in section 2.6.

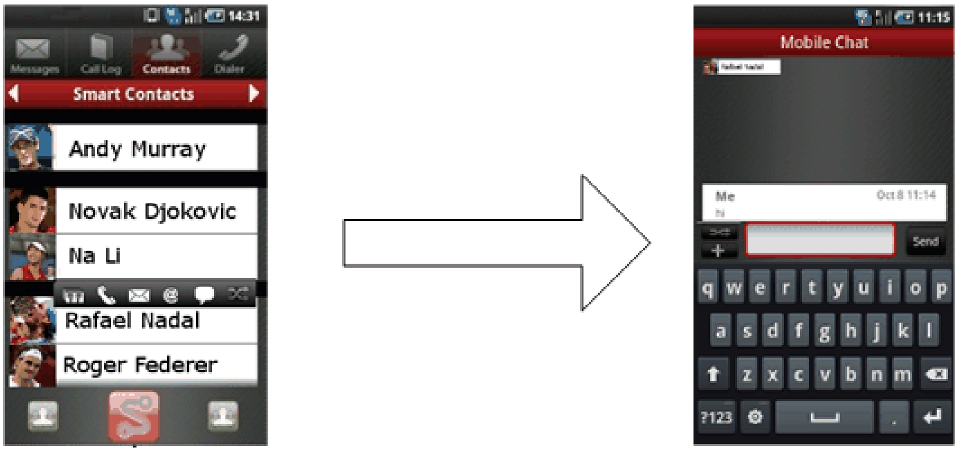

Figure 26: Reference UX for accessing chat from address book/call-log

2. Chat application: There should be a dedicated Chat application entry point in the device menu, task oriented initiation. This application will provide access to the chat history and gives the possibility to start a new chat.

Figure 27: Reference UX for starting a chat from the Chat application

Once the Chat application is opened, the user is presented with the complete list of RCS contacts with Chat capability. Whether or not contacts which are currently not registered are shown depends on the Chat store and forward policy (see IM CAP ALWAYS ON in Table 75) chosen by the Service Provider.

In addition to the “start a new chat” functionality, the Chat application allows the user to browse the Chat History, both 1-to-1 and Group Chat sessions:

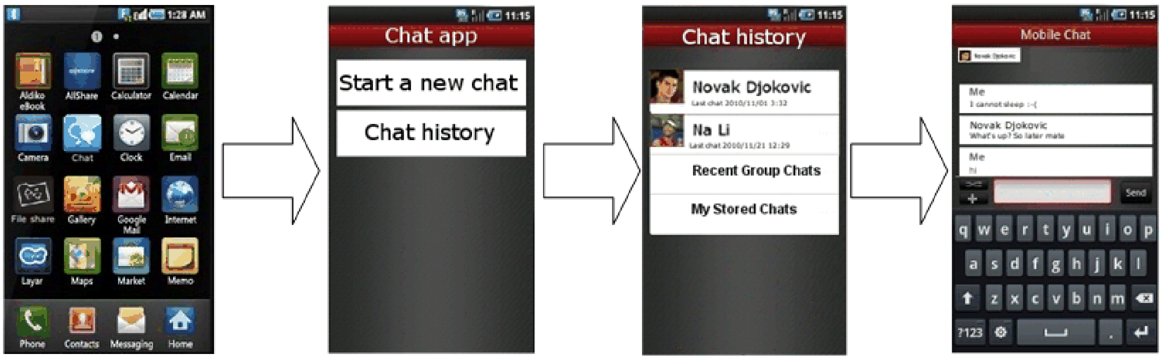

Figure 28: Reference UX for starting chat from the Chat application history

In this case, when the chat is started the last messages exchanged with that contact (or group of contacts) are shown even though the conversation might have been from another device. In the chat history, the user can also browse through chat sessions that he has selected for permanent storage (if the Common Message Store feature is available for the user) and start a conversation from those. The context of the past chat is not relayed to the other party/parties.

3. File Transfer (receiver): When transferring a file and with the aim of establishing a communication context for the transfer (the receiver may want to know for instance why the sender is sharing that file), after the transfer has been accepted the file transfer is presented to the receiver as a chat UX with a file being transferred.

NOTE: At the time the File Transfer request is presented, the chat session is not started; the chat session will only start when/if the receiver sends a chat message back to the sender.

Figure 29: Reference UX for File Transfer on the receiver side

From the UX point of view the requirements are summarised below:

- The user shall not be aware of the solution which is being used to transfer the file both at originator and terminating side.
  - Sender: There is no difference from a UX point of view between an accepted transfer and the upload to the HTTP content server. A progress bar is shown to track the progress.
  - Recipient: There is also no difference from a UX point of view in terms that the accept button is shown and if pressed the file is either downloaded from the sender (accepting the SIP INVITE) or from the content server (via HTTP/HTTPS)
- The only addition to the experience is the delivery notification that shall show the user when the file has been received by the other party (not stored in a middle-man server). In terms of UI, it is recommended to reuse the approach that is already in place for chat notifications so the UX is consistent among services.

Please refer to section 3.5 which covers the RCS File Transfer service in detail.

3.3.6.3 Initiating a chat

RCS User A initiates a chat by selecting one of his contacts User B from the address book, contact list or Chat application in one of his devices.

The device of User A determines whether User B is available to use the Chat service at that time, using one of the methods in section 2.6.

If User B is not available and there is no Chat store and forward Server on User A's side nor on User B's side, or no chat interworking to SMS/MMS, or if an answer to the query is not received in less than a time lapse (left to OEM User Experience criteria), then the contact is shown as ‘Not available for a Chat session’, and the SMS/MMS service or CPM Standalone Messaging service could be offered as a messaging option. Once the availability of the chat service is ensured end-to-end and User A performs the appropriate UI actions on the device, a message composer and an empty chat window are opened.

When User A types the first message and presses the “Send” button, device A will initiate a Chat session invitation toward B (for the multidevice scenario see multidevice handling in section 3.3.4.1.7).

The one or more devices of User B receiving a Chat session invitation, may either all be configured to auto-accept the invitation, or the devices may wait for user action before accepting the invitation. If a spam filter or a black list is implemented on any one of User B's receiving devices and User A is in the black list, the invitation is terminated following the procedure described in section 3.3.4.1.1.

On the User B side, a notification (UI dependent) is displayed on each receiving device to inform the user about the incoming message. The user is able to read the message and go to the chat window to answer the message on the device of his choice.

User A can type additional messages before the chat is answered, that is before the Chat session is established. On User B's side, a notification may be displayed for each received message (UI dependent). The sender's device may buffer the messages if the device is configured to wait for a Chat session to be set up before sending messages.

3.3.6.4 Answering a chat

There may or may not be an explicit acceptance of the user to answer a chat.

3.3.6.5 Messages exchanged in an established chat

Providing a Chat session is established, messages are exchanged between User A and User B. A delivery notification is requested for each message and a display notification is optionally requested.

The recommendation is to show the information received in the delivery and display notifications only within the Chat window without the need for a pop-up or information message when the user is outside of the Chat application.

3.3.6.6 Message display and message store

All messages are stored in the participating devices, together with a time indication and an appropriate indication of the sender and the receiver of each message. This time indication shall be obtained from the CPIM *DateTime* header for received messages. Since according to section 3.3.4.1 these values should be set by the Messaging Server, this allows for a correct time based indication for those messages without depending on the device's own clock which may not have been set correctly. For sent messages however the only clock available at transmission time is the device's own clock.

However, it is Messaging Server responsibility to deliver messages in the correct order, so the RCS Client is able to rely on the reception order to interleave the incoming and outgoing messages. Please note that the shown message time at the UX should be based on the network time (i.e. the CPIM DateTime header, when available) in order to correctly display the time of store and forwarded messages.

When a Common Message Store is available for the user, the messages are synchronised with the Message Store Server as specified in section 4.1.

When the storage limit is reached, deletion might occur on a first in/first out (FIFO) queue policy. It is open to OEM criteria how to implement other opt-in deletion mechanisms (e.g., ask always, delete always, delete any conversation/message from specific contacts, etc.).

3.3.6.7 Leaving the chat composing window

Once a 1-to-1 chat is established any of the two users can leave the composing window without closing the chat. For example, a user could move to his mobile home screen to check an incoming email, or make a phone call.

While the chat composing window is not shown (i.e. it is not the foreground window) any incoming message belonging to that chat will trigger a status notification (UI dependent) so the user is aware of the new message and may return to the chat composing window to answer it.

Also, the user could decide to return to the chat composing window and send a new message without receiving one. The user would be able to achieve this via the Chat application, which will display the ongoing chats, or via the address book by clicking on the contact with whom he is involved in the chat session.

In both cases, when the user returns to the chat composing window, all the messages are displayed.

3.3.6.8 'IsComposing' notification

When a user starts typing in the chat composition window and privacy settings allow it, an 'IsComposing' notification is sent to the other user. That user's UI will then display an indication in the chat composing window to indicate it (UI dependent).

The recommendation is to show the information received in the 'IsComposing' notification only within the Chat window without the need for a pop-up or information message when the user is outside of the Chat application.

The 'IsComposing' indication is removed from the UI when a new message is received, when a timeout occurs without receiving a new message, or when a new 'IsComposing' notification arrives.

3.3.6.9 Closing a chat / Re-opening a chat

Any of the two users can close the Chat session. This can be achieved from the chat composing window or from the Chat application.

The user should be able to re-open the chat. However, the resulting action at protocol level would depend on whether the Chat session is still open or not.

Closing the Chat session may not be notified to the remote user in the chat. At protocol level, the session is terminated. Therefore, if the remote user sends a message, a process similar to the initiation of a Chat session is performed as described in 3.3.6.3.

3.3.6.10 Re-Opening an older chat

An old chat conversation can be re-opened. From the user perspective, it is the same procedure as for initiating a chat (see section 3.3.6.3), except that when a message is sent, a new Chat session is established.

The device will then display the previously stored conversations with that contact preceding the current active one. If any displayed notifications still need to be generated, they are sent towards the original message sender.

3.3.6.11 User experience regarding notifications when several store and forward messages arrive in a short period of time

If a user has several chat messages waiting in storage in the Messaging Server to be delivered, the UX may be impacted if many Chat message notifications appear at the sender's device when the messages are delivered to the receiver after the receiving user gets registered again.

To avoid this situation and, specifically, when receiving stored and forwarded messages, the suggested experience follows:

- Only the delivery and/or display of the first message is shown in a notification to the sending user. The remaining store and forward messages are delivered but they do not cause a notification to be shown to the message sender.
- If messages from several sending users are received, only one message notification per user containing the first delivered message per sender is shown to the recipient user.

As mentioned previously this is a suggested guide and not mandated behaviour.

NOTE: The described behaviour refers to notifications shown on screen to the message recipient and does not affect the behaviour with regard to the sending of delivery notifications. Those are still sent for all received messages for which such a notification was requested.

3.4 Group Chat

3.4.1 Feature description

The Group Chat service enables users to exchange messages between many users instantly.

The following RCS features are described:

- Interworking of participants in a Group Chat to SMS/MMS
  This feature requires a Messaging Server to interwork the messages for participants without an RCS device to and from SMS or MMS.
- "Delivered" message disposition
  This allows the sender of a message to be notified when a message has been delivered to the recipient. These can be delivered inside and outside of the group chat.
- "Displayed" message disposition
  This allows the sender of a message to be notified when a message has been displayed on one of the recipient's devices.

NOTE: This notification cannot certify that the recipient has actually read the message. It can only indicate that the message has been displayed on the recipient's terminal User Interface (UI). These can be delivered inside and outside of the group chat.

- IsComposing indications
  This allows a user in a Group Chat conversation to see when another user is typing a new message/reaction.
- Local Black List
  The terminal/client may support a locally stored Black List to handle incoming Group Chat requests. Users are allowed to qualify undesired incoming Group Chat requests as spam. This prevents subsequent messages from those originators to be shown or even notified to the user. Also, this undesired traffic will not be acknowledged to have been read. The Black List behaviour applies not only to Group Chat but also to Chat and to File Transfer.
- Local Conversation History
  The terminal/client supports a locally stored conversation.
- Support for the Common Message Store
  The Common Message Store may be used to synchronize the messages between devices. It also allows the user to keep a back-up of important conversations in the network.
  In the device, alignment is expected between the local Conversation History and the synchronization with the Message Store Server.
- User Alias (Display Name)
  A user defined display name can be sent when initiating a communication with another user.
- Long lived Group Chat
  Once a user initiates a RCS Group Chat, any remaining participant shall be able to restart it, even if the Group Chat had been torn down by the Messaging Server because of inactivity.
- Store and Forward feature
  Messages missed because a participant has not yet joined the Group Chat or who was offline when the Group Chat started are stored and delivered when the participant becomes available or joins.
- Closed Group Chat
  A user initiating a Group Chat or Messaging Server can specify that a Group Chat shall be closed, meaning that no one is permitted to add participants to the Group Chat.
- Leaving a Group Chat
  For a Closed Group Chat, a user who explicitly leaves cannot rejoin.
  For a regular Group Chat, once that Group Chat terminates because of inactivity, a 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. If the Group Chat session is still ongoing, the user may rejoin even after he explicitly left.

A Group Chat can only be initiated by a user belonging to a Service Provider which has deployed a Messaging Server.

A Service Provider may disable the whole Group Chat functionality via the GROUP CHAT AUTH configuration parameter (see Table 75 in Annex A). In case Group Chat is disabled, the client shall not be able to either initiate or participate in a group chat session and reject any group chat session invitation that it receives.

A Closed Group Chat can be set up meaning no new participants may be added. If a Group Chat is closed, the Messaging Server indicates to all group participants that it is closed. A Service Provider may offer the choice to the user, or may only offer either one type of Group Chat or the other.

Once a participant explicitly leaves a Closed Group Chat by sending a SIP BYE request, it is not possible to rejoin since by definition it is not possible to add participants to a Closed Group Chat.

There is one type of Store and Forward for Group Chat. The Store and Forward feature, where messages are stored for a participant if he joins late, or never joins the group chat while it is ongoing, as well as if that participant has connectivity problems.

The Store and Forward feature for Group Chat is only available for a participant when his Service Provider deploys a Messaging Server.

Stored messages are delivered to the participant as follows:

1. When the user sends a request to rejoin the Group Chat. The Messaging Server will deliver the stored messages whether or not the Group Chat is still ongoing.
2. When the user accepts an invitation to join a Group Chat for which there are stored messages.

NOTE: If the user rejects a Group Chat invitation, all stored messages for that Group Chat are deleted.

3. Once the Group Chat is over or inactive and the Messaging Server knows the user is registered in IMS, or once the Messaging Server Participating Function receives an indication that the user has registered in IMS, the Messaging Server sends an invitation to the participant to deliver the Group Chat messages. If this invitation is not answered then further delivery attempts will be made based on local Service Provider policy. If none of the delivery attempts succeeds and the messages are not delivered, they will expire after an amount of time configured by the Service Provider. If the user rejects the invitation, all stored messages for that Group Chat are deleted.

3.4.2 Interaction with other RCS features

Interaction of Group Chat with other RCS features is described in section 3.3.2.

If the user wishes to transfer a file to Group Chat participants, this can be done using the procedure in section 3.5.4.2 when supported by the conference focus or in section 3.5.4.8. Otherwise the user’s device must do this by sending the file one by one to each Group Chat participant, and it may or may not appear in the Group Chat window.

3.4.3 High Level Requirements

The following list of high level requirements applies to Group Chat:

- Client devices:

3-4-1 "Delivered" notification request and response
3-4-2 "Displayed" notification request and response

NOTE: The client device should allow the user to enable or disable the displayed notifications request and response.

3-4-3 Delivery of notifications (delivered and displayed) outside a session
3-4-4 IsComposing indications
3-4-5 Procedures associated to the store and forward of both messages and notifications performed by the Messaging Server
3-4-6 Ability to request a regular Group Chat or a closed Group Chat
3-4-7 Ability to restart a Group Chat conversation that is idle

- Messaging Server: In addition to the above requirements:

3-4-8 The Messaging Server may provide interworking of Group Chat to SMS/MMS
3-4-9 The Messaging Server shall provide store and forward of both messages and notifications
3-4-10 Store and forward is a function which is provided on the terminating MNO network.
3-4-11 A Messaging Server hosting the conference focus shall be able to put the Group Chat in an idle state after a Service Provider's defined inactivity period. When closing a Group Chat session because it is idle a Messaging Server hosting the conference focus shall store the session identity and the current participant list of the chat allowing participants to restart the conversation.
3-4-12 The Messaging Server may provide the ability to set up a regular Group Chat or a closed Group Chat and shall indicate to all the participants whether the Group Chat is closed or not
3-4-13 A Messaging Server hosting the conference focus shall allow any participant in the stored participant list to restart an idle Group Chat. The restarted Group Chat shall keep the same session identity and the same participant list.

3.4.4 Technical Realisation

Group Chat technical realisation is based on the "Ad-Hoc Session Mode messaging" as described in [RCS-SIMPLEIM-ENDORS] and in [RCS-CPM-CONVFUNC-ENDORS], (depending on the setting for CHAT MESSAGING TECHNOLOGY defined in Table 75 in Annex A).

Support for delivery and display notifications within a Group Chat is added to the functionality endorsed in [RCS-SIMPLEIM-ENDORS] and [RCS-CPM-CONVFUNC-ENDORS]. For OMA CPM, also the functionality to support sending "IsComposing" messages within a Group Chat is added.

The Closed Group Chat feature is provided by a Messaging Server, and all participants in such a Group Chat are made aware that it is closed.

The Store and Forward feature for Group Chat is provided by each Group Chat participant's own Messaging Server.

To prevent revealing the user identity when transmitted over unprotected links, the client should set the CPIM To header of a Message exchanged in a Group Chat to *sip:anonymous@anonymous.invalid*. For Delivery and Display notifications, it will be set as described in section 3.4.4.1.5. As the CPIM From header is needed to identify the sender of the message the user's identity will be provided there and include the display name.

The originating Messaging Server shall always set the CPIM DateTime header in the chat messages it receives. The originating Messaging Server shall also set the CPIM DateTime header and IMDN DateTime element in notifications. In both cases, the Messaging Server shall overwrite any DateTime information provided by the client. A client receiving these requests should therefore rely on these headers containing the correct time rather than on locally available time information.

To allow a user to rejoin or restart a Group Chat, the user's client is required to know the actual focus Session Identity created by the Messaging Server when the Group Chat was initiated. The recommended approach is that any SIP proxy (e.g. Interworking Call Session Control Function [I-CSCF], P-CSCF, IMS-ALG, Session Border Controller [SBC], NAT) in the path between the Messaging Server and the client transparently forwards a received Contact header field from the network being sent towards the UE when the Contact header field contains the "isfocus" feature tag. This is as specified in sections 5.2.7.2, 5.2.7.3, 5.7.5.1 and 5.10.5, of [3GPP TS 24.229].

In normal circumstances, for a given RCS Group Chat ID there is at most only a single session is active at a time for an RCS user. In the OMA CPM realisation, as detailed in [RCS-CPM-CONVFUNC-ENDORS], only one Group Chat is supported within a given CPM Conversation.

3.4.4.1 Technical Realisation of Group Chat with Delivery and Display Notifications

3.4.4.1.1 Initiating a Group Chat

User A initiates a chat by selecting some of his contacts (Users B, C and so on, up to a limit set by the OMA SIMPLE IM parameter MAX_AD-HOC_GROUP_SIZE – see Annex A) from the address book or from the Chat application in his device, or from the Contact List from the Broadband Access PC client. This choice may be offered only among the contacts known by his devices to be RCS users with Chat capability. It may be offered for all contacts if a Group Chat interworking service to SMS/MMS is available from the Service Provider (See configuration parameters IM CAP NON RCS GROUP CHAT, IM CAP NON RCS LIMIT GROUP CHAT and GROUP CHAT BREAKOUT ALLOWED PREFIXES, in Table 75 in Annex A).

If the IM CAP NON RCS GROUP CHAT is disabled, then the device recognises whether the Group Chat service is available for a particular contact by using the service capability exchange via Presence or OPTIONS as described in of section 2.6.

When initiating a Group Chat User A may first provide a subject for the conversation that will be provided to all invitees of the group chat. To provide a good user experience, it is recommended to provide it even to those that are invited later when the chat is ongoing. When User A presses the "Send" button, device A initiates a Chat session with the Messaging Server by sending a SIP INVITE request to the conference factory (carrying the subject in the Subject header if one was provided) with a new RCS Group Chat ID. The Messaging Server sends SIP INVITE requests to the other participant users indicated in a recipient-list body in the INVITE request received from User A. The list of invited participants is sent in the Group Chat invitation, and is also sent out to all invited participants.

When a user's client receives a Group Chat invitation from the Messaging Server, the user may accept or reject the invitation. Alternatively the user's client may auto-accept[29] a Group Chat invitation, depending on a configuration parameter IM SESSION AUTO ACCEPT GROUP CHAT as defined in Annex A. A client on which Group Chat has been disabled via the GROUP CHAT AUTH configuration parameter (see section A.1.4.3) shall not notify the user and automatically reject any received SIP INVITE request for a Group Chat with a SIP 488 NOT ACCEPTABLE HERE response.

When at least one invited participant accepts the invitation, the 200 OK response is sent back to User A and the Group Chat is set up. At that moment, User A (or the user that accepted) can write a first message in the chat. Any messages sent to the focus during this start up phase before a final response is received from each invited participant are temporarily queued in the Messaging Server Controlling Function until there is a final response from all invited participants. Depending on each participant's response:

- A 200 OK is received, the focus will send all the messages temporarily queued;

When all the final responses have been received, the focus will stop queuing messages.

After acceptance, the client shall subscribe to the conference event package to retrieve the list and status of the users in the Group Chat.

User A's device shall also subscribe to the conference event package. The SUBSCRIBE request shall be routed to the Messaging Server Participating Function to allow it to participate in the dialog. When the client receives the resulting SIP NOTIFY requests carrying the conference state information, the identity of each user shall be matched against the Contact List in the device to present a user friendly name. If a user is not found in the Contact List, the display name provided in the conference state, if any, for that user should be used. As a complement to [RCS-SIMPLEIM-ENDORS] and [RCS-CPM-CONVFUNC-ENDORS], the notifications pertaining to the conference event package should convey information about the pending participants (i.e. the "*pending*" state is used in the <*status*> element of the corresponding endpoint as per [RFC4575]).

NOTE: To avoid sending notifications to participants twice in short succession, the conference focus shall briefly delay notifying the existing participants of the "pending" state of the newly added participant to allow for automatic acceptance of the Chat (e.g. because of Store and Forward as described in section 3.4.4.3). In that case the participant's state will change to "active" almost immediately.

The Messaging Server will open sessions to Users A, B, C and so on, up to a configured limit which should be set to the same OMA SIMPLE IM parameter value configured in the clients, i.e. MAX_AD-HOC_GROUP_SIZE.

In the user interfaces of the receivers' client on which the GROUP CHAT AUTH configuration parameter is enabled (see section A.1.4.3), a notification (UI dependent) shall

[29] Note that the Service Provider multidevice policy has to be consistent with the Group Chat auto-acceptance policy.

be displayed to inform the user about the incoming invitation. This notification should clearly state that it is an invitation to a Group Chat making the users aware of this fact and should take into account the IM SESSION AUTO ACCEPT GROUP CHAT configuration parameter defined in section A.1.4.3. This notification can also indicate the other users that were invited as well as the subject of the conversation if one was provided.

The supported content types in the SDP exchanged in the SIP INVITE request and the associated 200 OK response shall be indicated as follows:

- In the SDP of the SIP INVITE request and response, the a=accept-types attribute shall include only *message/cpim*, i.e., "*a=accept-types:message/cpim*".
- Multimedia content within a Chat session is not permitted. Therefore, in the SDP of the SIP INVITE request and response, the *a=accept-wrapped-types* attribute shall only include *text/plain*, *message/imdn+xml* and *application/im-iscomposing+xml.* If File Transfer using HTTP is supported (see section 3.5.4.8) then the a=accept-wrapped-types attribute shall also include *application/vnd.gsma.rcs-ft-http+xml.* If Geolocation PUSH is supported (see section 3.10.4.1.2), then the *a=accept-wrapped-types* attribute shall also include *application/vnd.gsma.rcspushlocation+xml*. To transfer multimedia content during a chat, File Transfer is used.

NOTE: During session setup the client and the conference focus shall also take into account the procedures described in section 3.5.4.2 and when they support File Transfer via HTTP in the Group Chat also those in section 3.5.4.8.1. When they support Geolocation PUSH in the Group Chat they shall also take into account the procedures in section 3.10.4.1.2.

Once the Group Chat has been initiated, the focus Session Identity uniquely identifying that Group Chat may be kept by the original initiator's Messaging Server Controlling Function either for an amount of time configurable by the Service Provider, or until the original initiator's Messaging Server Controlling Function no longer authorises other users to restart this Group Chat (e.g. because of a Service Provider policy when the initiator explicitly left the chat). Depending upon service provider policies, this focus Session Identity can then be used by any of the participants in the Group Chat to restart it by simply attempting to rejoin the Group Chat as described in section 3.4.4.1.7.

3.4.4.1.2 Adding participants to a Group Chat

Once a Group Chat is established, the local Service Provider policy decides whether only the initiator is allowed to add participants to the Group Chat or whether any participant is allowed to add more participants. A Service Provider may choose to have a local policy that allows participants that are their own subscribers to add participants, but participants from other Service Providers would not be allowed.

The maximum number of participants 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] when the client subscribes to the conference event package. Participants may be added provided the maximum-user-count is not reached or the focus's Service Provider policy allows it. If these values are not present in the conference event package then that the MAX_AD-HOC_GROUP_SIZE configuration parameter may be used instead.

Participants on the participant list can be added again (re-invited), provided that they are not in the active or pending state or have left the chat involuntarily. This way participants that have not accepted the INVITE request before it has timed out, or participants that left the chat voluntarily (see section 3.4.4.1.3.1) can be added again.

For a Closed Group Chat (see section 3.4.4.2), no one can add participants. If the Group Chat is a Closed Group Chat, the Messaging Server will return an error response to the SIP REFER request. If adding participants fails because of one of the reasons above, it is expected that the Messaging Server's error response include a Warning header and appropriate explanatory text as per the [RCS-SIMPLEIM-ENDORS] or [RCS-CPM-CONVFUNC-ENDORS] (depending on the setting for CHAT MESSAGING TECHNOLOGY see Table 75 in Annex A).

When adding participants, as a clarification to [RCS-SIMPLEIM-ENDORS] or [RCS-CPM-CONVFUNC-ENDORS] (depending on the setting for CHAT MESSAGING TECHNOLOGY see Table 75 in Annex A), the client shall :

- Include an RCS Group Chat ID in the REFER request set to the RCS Group Chat ID of the pertaining Group Chat,
- Include a Subject header in the REFER request set to the pertaining Group Chat subject, if the pertaining Group Chat was created with a subject.

3.4.4.1.3 Closing Group Chat

A SIP/MSRP session established between one of the user's devices and the network allowing a user to participate in a Group Chat will be closed for either one of the following reasons:

- The user explicitly indicates that he's not willing to take part in the Group Chat anymore (see section 3.4.4.1.3.1),
- An error condition occurs preventing a particular device from further maintaining an active session that allows a user to participate in a Group Chat (see section 3.4.4.1.3.2),
- Based on local service provider policies, the Controlling Function no longer wishes to maintain the active sessions of an ongoing Group Chat and therefore, forces the closing of the remaining active sessions (see section 3.4.4.1.3.3).

3.4.4.1.3.1 Explicit Departure

Any of the participants can voluntarily leave the Chat. When leaving voluntarily, a conversation history will exist in the user's device history with the messages associated with the chat up to the point the user left.

When a participant indicates their desire to voluntarily leave the Group Chat session their device shall send a SIP BYE request that includes a *Reason* Header field (as defined in [RFC3326]) with the protocol set to *SIP* and the protocol-cause set to *200* (e.g. SIP;cause=200;text="Call completed"). In addition, the client shall unsubscribe from receiving the Chat participant information. When it receives a SIP BYE request the Controlling Function will convey with a new conference state event package notification to the remaining participants and in case of voluntary departure, remove the user from the participant list. In this new conference state event package notification the Controlling

Function shall include additional elements and values defined in [RFC4575] to indicate why the participant is not taking part in the conversation any longer. If the participant left voluntarily (that is, the SIP BYE request included a *Reason* header field with the protocol set to *SIP* and the protocol-cause set to *200*), the conference focus shall indicate the departed participant‘s status as “disconnected” and include a disconnection-method element the value of which shall be set to “departed”.

When the chat is (re)started, any participant can also leave or decline the Group Chat by rejecting the Group Chat invitation with a 603 Decline response. When receiving a SIP 603 Decline response to a SIP INVITE request for a Group Chat, the Controlling Function will remove the participant that declined from the participant list and convey a new conference state event package notification to the remaining participants. In this new conference state event package notification, the controlling function shall set the departed participant’s status to “disconnected” and include the disconnection-method element set to a value of “failed” with a reason sub-element set to code *603 (e.g. <reason>SIP;cause=603;text="Decline"</reason>)*.

NOTE: When the user explicitly leaves a regular Group Chat (i.e. as opposed to a Closed Group Chat as defined in section 3.4.4.2), their client may store the Group Chat’s IM Session identity for some time to offer the user the option to rejoin. If the user makes use of this option the device shall handle this according to the procedures described in [RCS-SIMPLEIM-ENDORS] or [RCS-CPM-CONVFUNC-ENDORS]. 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 an attempt to rejoin results in a SIP 404 Not Found response, the chat should be considered to be no longer active.

In case the user wants to leave a Group Chat at a time where the Group Chat is idle, the RCS client shall restart the Group Chat using the procedure in section 3.4.4.1.7 and shall send a SIP BYE request that includes a Reason Header field (as defined in [RFC3326]) with the protocol set to SIP and the protocol-cause set to *200* (e.g. SIP;cause=200;text="Call completed"). In this case, the client shall not subscribe for receiving the Group Chat participant information.

When User C leaves the Group Chat voluntarily, the other users that have subscribed to the conference focus 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 participants.

An RCS client receiving a notification of a participant leaving the Group Chat voluntarily, either by closing (i.e. using a SIP BYE request that includes a *Reason* header field with the protocol set to *SIP* and the protocol-cause set to *200*) or rejecting the Group Chat session invitation with a SIP 603 Decline response shall remove the participant from the locally stored participant list associated with the Group Chat.

A participant voluntarily leaving the Chat is removed from the Group Chat participant list used for restarting the chat as specified in section 3.4.4.1.7 by both the Controlling Function and the clients receiving a notification indicating this voluntary departure.

A participant of a Group Chat for whom the service provider removes the IMS identity and profile should depart from the Group Chat. In absence of triggers from the Participating Function in this case, the Controlling Function shall implement the following procedure. If the Controlling Function re-starts a Group Chat as defined in section 3.4.4.1.7 and it receives a SIP 404 Not Found response to the INVITE sent to a participant, it shall remove the corresponding participant from the participant list. The clients of the other participants in the Group Chat are informed via the notification of conference events within their subscription to the conference event package.

The elements of the removed participant's user endpoint in the conference event package shall be set as follows:

- Status of the user endpoint element is set to "*disconnected*",
- Disconnection-method of the user endpoint element is set to "*departed*".

3.4.4.1.3.2 Involuntary departure

A user can also leave the session involuntarily (e.g. due to loss of connectivity or other error situations). In this case, if it is still capable of doing so, the client or otherwise, the network element detecting the error situation should generate a SIP BYE request that includes a *Reason* Header field with the protocol-cause set to a value other than *200* (i.e. the protocol-cause must not be the value used for voluntary departure in section 3.4.4.1.3.1). It is recommended to use a *Reason* header field with the protocol set to *SIP* and the protocol-cause set to *503* (e.g. SIP;cause=503;text="Service Unavailable") as was defined in [3GPP TS 24.229] for bearer loss detected by the P-CSCF also for other network elements.

When a SIP BYE request carrying a *Reason* header field with a protocol-cause other than *200* is received by the Participating Function, the Participating Function will not forward the SIP BYE to the Controlling Function. The disconnection is transparent to Controlling Function and the participants.

NOTE: As it cannot be guaranteed in general that all network elements detecting an error situation will include a Reason header field nor that there are no legacy clients connecting to the network that do not include a Reason header field, it is recommended that the Messaging Server allows for a Service Provider policy controlling whether a received SIP BYE request without Reason header field is handled as a voluntary or involuntary departure. If such a policy is not provided, it is left to implementation.

3.4.4.1.3.3 Closing of the Group Chat

The remaining sessions for a Group Chat are closed by the Controlling Function amongst others when in following cases:

1. Less than the minimum number of active 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, e.g. if the originator leaves the Group Chat.

NOTE: 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).

When closing these sessions, a conference focus shall keep the focus Session Identity and associated information (e.g. participant list) in following cases:

- Case 1 above, when there are users on the participant list that have left involuntarily (see section 3.4.4.1.3.2).
- Case 2 above.

The conference focus shall maintain the information for at least one month.

NOTE: It is recommended that the Participating Function providing store and forward functionality for Group Chats as described in section 3.4.4.3 stores deferred Group Chat messages for at most one month to ensure that the original focus can be used should there be a need to restart the session as a consequence of the forwarding of the deferred messages.

The Messaging Server shall keep the focus Session Identity. This allows any of the participants that still were in the Group Chat when the session was closed to send a rejoin request that will result in the Group Chat being restarted. In that case, the Messaging Server also keeps the type of Group Chat (e.g. Closed) and the list of participants that were present when the Group Chat was torn down, and uses that list to check who is authorised to restart the Group Chat, and then to invite those participants when it receives the rejoin request from an authorised participant as described in section 3.4.4.1.7. When the Session Identity is maintained in case 2 above or in case 1 above for the situation where some of the participants left the session involuntarily or didn’t explicitly accept or reject the invitation, the Controlling Function shall include a *Reason* header field with the protocol set to *SIP* and the protocol-cause set to *480* (e.g. SIP;cause=480;text=”Bearer unavailable”) in the SIP BYE request that it sends to the remaining participants. In other scenarios where the messaging server keeps the focus Session Identity, it shall include a *Reason* header field, but that may relay different values.

For other cases where the Messaging Server no longer keeps the focus Session Identity for the Group Chat, any future attempt by a user to join or restart the Group Chat identified by the focus Session Identity will fail. The Controlling Function shall indicate this by including a *Reason* header field with the protocol set to *SIP* and the protocol-cause set to *410* (e.g. SIP;cause=410;text=”Gone”) in the SIP BYE request that it sends to the remaining participants. A client receiving a SIP BYE request including a *Reason* header field with the protocol set to *SIP* and the protocol-cause set to *410* may take this into account when restarting the Group Chat (see section 3.4.4.1.7) and avoid sending a rejoin request to the focus Session Identity.

Even if the focus indicated that it keeps the Session Identity and the participant list (i.e. the BYE request received by the clients included a Reason header with the protocol set to SIP and the protocol-cause set to a value different than 410), the clients participating in a Group Chat shall also keep the latest participant list. If a rejoin to an inactive Group Chat fails (e.g. because the Group Chat has been inactive for a very long time), the client can then restart the Group Chat as a new Group Chat using this latest participant list from the last Group Chat as described in section 3.4.4.1.7.

To avoid that users are removed from an ongoing session, the idle time during a Group Chat should be monitored by the Controlling Function ensuring that the session is torn down for all users at the same time. Unlike the situation for a 1-to-1 Chat where the Participating Function may tear down the session also in normal circumstances, any idle session monitoring on the Participating Function, if provided, shall therefore use timeouts that are significantly larger than the timers used in the controlling function. To allow configuring these Participating Function timers for intervening in error situations (e.g. loss of connectivity between the Participating and the Controlling Function), the maximum allowed idle time on the Controlling Function shall not be larger than 300 sec (i.e. 5 min). Because they are not relayed to all Participating Functions involved in the Chat, the Controlling Function shall not take the messages relayed over MSRP for delivering disposition notifications into account when monitoring the idle time.

NOTE: Whether to provide idle time monitoring of a Group Chat in the participating function for intervening in error situations is an implementation decision.

3.4.4.1.4 Chat message size limitations

This maximum size is controlled through the MAX SIZE IM configuration parameter defined in Table 75. Endpoints and the Messaging Server are expected to make use of the SDP attribute a=max-size to indicate the maximum message size to participants during session negotiation.

From the user experience perspective and assuming that the message size limitation is in enabled (i.e. the negotiated values are non-zero):

- For originated chat message the client shall not send the message bigger than the negotiated max-size value.
- If the user attempts to send a message larger than the negotiated limit, the message is not sent, and the user should be informed that messages of that size cannot be sent in the Group Chat session.

3.4.4.1.5 Delivery and Display notifications within Group Chat

Each message sent within a Group Chat may request a delivery notification and may request a display notification, similar to the previously described 1-to-1 chat (see section 3.3.4.1).

The recipient client generates the delivery or display notifications as described for 1-to-1 chat (see section 3.3.4.1), with the difference that the CPIM *To* header shall be set to the identity of the sender of the message, found in the CPIM *From* header of the incoming message, instead of to an anonymous URI. The identity to be used from the CPIM *From* header is the URI for the sender.

This requires that the Messaging Server support Private Messages within Group Chat.

If there is no on-going Group Chat in which to send these notifications, they shall be sent using SIP MESSAGE.

3.4.4.1.6 Interworking to SMS/MMS

For a Group Chat the behaviour for interworking to SMS/MMS is similar to interworking of a 1-to-1 Chat described in section 3.3.4.1.6 with the same entities being involved. The only difference being that the decision to interwork may be taken by the Controlling Function of the Messaging Server based on the same criteria used by the Participating Function for the case of a 1-to-1 session (e.g., based upon error), or by the terminating Participating Function (e.g. based upon error or Service Provider policy). Furthermore, as described in [RCS-CPM-IW-ENDORS] and [RCS-3GPP-SMSIW-ENDORS], the IWF will subscribe to the participant information and use that information to inform the SMS or MMS user of who also is taking part in the Group Chat.

NOTE: Messages sent during any interval that a participant becomes unavailable or loses connectivity during the Group Chat will only be stored for that participant due to the Store and Forward feature being available for that participant. See section 3.4.4.3 for more information on the Group Chat Store and Forward feature.

3.4.4.1.7 Restarting a Group Chat

When a Group Chat has been closed due to inactivity, it may be restarted at any time by any of the participants. A client shall not restart an already established Group Chat.

In order to restart a Group Chat, the RCS client shall try to rejoin using the focus Session Identity and same RCS Group Chat ID of the previous Group Chat session.

When the Messaging Server Controlling Function receives an incoming SIP INVITE request with a focus Session Identity for a Group Chat that is not in progress, it checks whether it still has the focus Session Identity along with the last participant list. If so, it checks whether the user is authorised to restart the Group Chat identified by the focus Session Identity in the SIP INVITE request, and if so, restarts the Group Chat using the associated participant list and as a Closed or regular Group Chat depending on the type it was before.

The participant list shall include the active participants (i.e. the list shall also include the participants in the pending state).

If the Messaging Server Controlling Function does not recognize the focus Session Identity, it shall return a SIP 404 error response. In this case, the RCS client shall initiate a new Group Chat as per section 3.4.4.1.1 re-using the same RCS Group Chat ID and with the latest participant list it has available for the Group Chat.

If the focus Session Identity is available but the user is not authorised to restart the Group Chat, the Messaging Server shall return a SIP 403 error response including a warning header set to “127 Service not authorized”. In this case, the RCS client shall initiate a new Group Chat as per section 3.4.4.1.1, using the last participant list it has stored to build the URI-list in the SIP INVITE request.

NOTE1: No specific handling for Group Chat restarts is required for non-2xx SIP responses other than those mentioned in this section.

For clients re-invited to an existing Group Chat the following procedures apply:

- If the client receives a Group Chat invitation with an RCS Group Chat ID the client has left voluntarily or it has been closed before, it shall apply the procedure for a new Group Chat invitation as defined in section 3.4.4.1.1.
- If a Group Chat invitation is received with the same RCS Group Chat ID of an already established Group Chat, the RCS device will auto accept the new Group Chat session.

3.4.4.1.7.1 Race conditions

Since a Group Chat can be restarted by two participants simultaneously, race-conditions exist between rejoin requests coming from the client and SIP INVITE request originated by the Controlling Function. As the middle element, most of these situations will be detected by the Messaging Server Participating Function that shall handle these situations as follows:

1. For the case where the Messaging Server Participating Function receives an incoming SIP INVITE request from the client for a Group Chat for which a SIP INVITE request was already sent (matching shall be done based on the focus Session Identity or the RCS Group Chat ID) (i.e. the INVITE requests have crossed between the client and the Participating Function):

    a) If no session is established yet with the Controlling Function (see also section 3.4.4.3), the Messaging Server Participating Function shall forward this INVITE request from the client to the conference focus and handle the SIP INVITE request from the client as a regular B2BUA with this session setup to the Controlling Function.
    b) Otherwise the Messaging Server Participating Function shall accept the SIP INVITE request from the client, establish the MSRP channel and forward any messages and notifications received from the client in the already established MSRP session with the Controlling Function.

    Messages and notifications received from the Controlling Function shall only be forwarded in the MSRP channel to the client that was last to be established. Based on local policy, the Messaging Server Participating Function shall terminate the unused session by sending in the corresponding SIP dialog a SIP BYE request carrying a Reason header field with the protocol set to SIP and the protocol_cause set to 480 (e.g. SIP;cause=480;text="bearer unavailable").

    This means, that in all cases where such a race condition occurs temporarily, two sessions are established between the client and the Participating Function and only one between the Controlling Function and the Participating Function. Between the client and Participating Function only the MSRP session that was last to be established shall be used.

2. For the case where the Messaging Server Participating Function receives an incoming SIP INVITE request from the Controlling Function for a Group Chat for which a SIP INVITE request was already sent (matching shall be done based on the focus Session Identity or the RCS Group Chat ID) to the Controlling Function (i.e. the INVITE requests have crossed between the Controlling Function and the Participating Function), the Messaging Server Participating Function may either

a) Forward this INVITE request to the client, or
b) Accept both the session from the Controlling Function and the rejoin request from the client and link both dialogs as a B2BUA. In that case when the Controlling Function accepts the INVITE request, the Participating Function shall establish the MSRP session and only use the last MSRP channel to be established until either the Controlling Function closes one of the sessions, closes the entire chat or the user leaves the Chat. In the last case the Messaging Server Participating Function shall send the corresponding SIP BYE request in both sessions.

Also, the Controlling Function shall accept a rejoin request received from a participant for which there was an outstanding INVITE request. It shall ensure that only one session is used and that messages from the participant are not returned in the other session. To achieve this, it is recommended to only send messages in the last MSRP channel to be established. Once it has received messages or notifications over that connection it may close the other session by sending in the corresponding SIP dialog a SIP BYE request carrying a Reason header field with the protocol set to SIP and the protocol_cause set to 480 (e.g. SIP;cause=480;text="bearer unavailable").

3.4.4.1.8 Multidevice handling and the Common Message Store

Multidevice handling occurs when a user has more than one device (e.g., PC and mobile). When a new Group Chat is initiated and an invited user, User B, has multiple devices registered at the same time, the Group Chat session request shall be forked to each device for that user by the terminating Messaging Server Participating Function. Forking at the Messaging Server is further described in 3.4.4.1.8.1.

When a user wishes to restart a chat on a different device, it may not have the latest information on the Group Chat. In particular, it will not have the focus Session Identity for the Group Chat if it did not receive the initial SIP INVITE request. This would occur if the device had not been registered in IMS when the initial INVITE request was sent out.

When a Common Message Store is available for the user, the Group Chat messages are stored in the Common Message Store as they pass through the user's Participating Function. If a network initiated SIP BYE request is received, then in addition to storing the Conversation-ID (if present) and Contribution-ID, the focus Session Identity for the Group Chat, whether it was closed or regular, and a message containing the current list of participants need to be stored. This implies that the Messaging Server Participating Function is required to keep track of the list of participants for the Group Chat by subscribing to the participant information and keeping the relevant information received in the corresponding SIP NOTIFY requests. The messages are synchronised with the Message Store Server as specified in [RCS-CPM-MSGSTOR-ENDORS].

3.4.4.1.8.1 Forking on the Messaging Servers

The forking procedures for Group Chat are the same as those described for 1-to-1 chat in section 3.3.4.1.7.1.

Additionally, each device shall subscribe to the conference state event package, once it accepts the INVITE. The Participating Function serving that user shall make sure that there is only one subscription maintained per user towards the Controlling Function.

If a client indicates voluntary departure from the Group Chat when there is no active device as per section 3.4.4.1.3.1, the terminating Participating Function shall close the session towards the other clients and indicate to the Controlling Function that the user has left the group chat.

If a client leaves a Group Chat involuntarily from one of their devices as per section 3.4.4.1.3.2 and there is no user device that has accepted the Group Chat, the Messaging Server shall continue forking incoming messages to the rest of the user devices.

3.4.4.1.9 Group Chat Service Identification

The RCS client shall populate the P-Preferred-Service header field in all CPM or SIMPLE IM requests with the CPM Feature tag defined for the service, as described in [RCS-CPM-CONVFUNC-ENDORS]. The S-CSCF or AS that performs the service assertion in the originating network shall add the P-Asserted-Service header field set to the value of the asserted CPM service ICSI (i.e. “urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group” for Group Chat, or “urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred” for deferred delivery done as part of the Store and forward realisation) and remove the P-Preferred-Service header field before further routing the request.

NOTE: During a transition period towards full compliance, the network support for asserting the service is recommended but not mandatory.

A receiving network element and RCS client should ignore any SIP header fields that they do not understand (e.g. P-Preferred-Service, or P-Asserted-Service header fields).

3.4.4.2 Technical Realisation of a Closed Group Chat

In order for a device to request that a particular group chat remain closed to the addition of new participants, the device sets the *a=chatroom* attribute defined in [RFC7701] in the SDP of the SIP INVITE request with the CPM reserved chat-token value, as described in [RCS-CPM-CONVFUNC-ENDORS], to indicate the RCS Closed Group Session in the SIP INVITE. The SDP attribute value shall be:
a=chatroom:org.openmobilealliance.groupchat.closed;

If a Service Provider only wishes to provide Group Chats that are closed, then even if the SDP offer did not include the *a=chatroom* attribute, a Closed Group Chat is created and the *a=chatroom* attribute is returned to the sender in the SDP answer in the 200 OK to the Group Chat initiator.

For a Closed Group Chat, the *a=chatroom* attribute (i.e. “*a=chatroom:org.openmobilealliance.groupchat.closed”*) is placed in the SDP offer in the SIP INVITE request sent to invited participants:

A device recognising this attribute may disable the possibility of adding participants to the Group Chat. A device not recognizing this attribute shall ignore it, but any attempt by the device to add participants will result in an error being returned from the Messaging Server.

3.4.4.3 Technical Realisation of Store and Forward functionality for a Group Chat Participant

A Messaging Server may provide Store and Forward functionality for users participating in Group Chat. This feature applies no matter whether this Service Provider or another Service Provider is hosting the Group Chat.

3.4.4.3.1 Initiating a chat

Section 3.4.4.1.1 applies. When a user who is offline is invited to a group chat or a user does not respond to a SIP INVITE request for a group chat (i.e. the request times out), the participating function will accept the group chat on behalf of the user and subscribe to the conference state information for that chat session. If it can handle File Transfer via HTTP, the participating function shall include the File Transfer via HTTP IARI in the Contact Header of the SIP 200 OK response sent when accepting the session and if no wildcard was used, also the *application/vnd.gsma.rcs-ft-http+xml* shall be included in the *a=accept-wrapped-types* SDP attribute that is included in that SIP 200 OK response. If it can handle Geolocation PUSH content, the participating function shall include the Geolocation PUSH IARI in the Contact Header of the SIP 200 OK response sent when accepting the session and if no wildcard was used, also the *application/vnd.gsma.rcspushlocation+xml* shall be included in the *a=accept-wrapped-types* SDP attribute that is included in that SIP 200 OK response.

Given that unless the user declines the Chat a Messaging Server Participating Function will accept the session, the Messaging Server Participating Function should anticipate to this and immediately accept the session from the controlling function without waiting for client's final response first. This behaviour will ensure that all messages sent in the Group Chat will ultimately reach the user also in exceptional circumstances where the Chat is closed by the Controlling Function before a final response was received from the user. When accepting the session before there is a final response from the user, the Messaging Server Participating Function shall in case a SIP 603 Decline response is received from the user terminate the session by sending a SIP BYE request to the Controlling Function carrying a Reason Header Field with the protocol set to *SIP* and the protocol cause code set to *200* (e.g. SIP;cause=200;text="Call completed").

3.4.4.3.2 Adding Participants to a Group Chat

The text in section 3.4.4.1.2 applies. If the Group Chat is a Closed Group Chat, the Messaging Server will return an error response to the SIP INVITE request.

Messages exchanged in the Group Chat before the new participant is invited are not subject to the Store and Forward functionality for that participant.

3.4.4.3.3 Closing Group Chat

The text in section 3.4.4.1.3 applies.

When the user explicitly leaves the Group Chat, the Messaging Server Participating Function shall, as well as the forwarding of this action to the Controlling Function (as described in section 3.4.4.1.3), also discard any remaining messages and notifications related to this Group Chat that were stored for delivery to that user.

3.4.4.3.4 Chat messages size limitations

The text in section 3.4.4.1.4 applies.

3.4.4.3.5 Delivery and Display notifications within Group Chat

The text in section 3.4.4.1.5 applies.

3.4.4.3.6 Interworking to SMS/MMS

The text in section 3.4.4.1.6 applies.

3.4.4.3.7 Restarting a Group Chat

The text in section 3.4.4.1.7 applies.

3.4.4.3.8 Storing messages for a participant who loses connectivity

If a Messaging Server implements the Group Chat Store and Forward feature, then if a Messaging Server Participating Function serving a Group Chat participant detects that the participant is unreachable (e.g. loss of network coverage), it becomes an endpoint for the Group Chat. In this case, it shall store any messages and disposition notifications (but not isComposing Notifications) received for that participant and keep the dialogs for the Chat session alive by sending timely refresh requests as defined in [RFC4028]. This way, the conference focus continues to keep the participant in the participant list for the Group Chat. Furthermore the Messaging Server Participating Function shall ensure that it has a subscription to the conference state for the duration of the Group Chat, or until the user re-joins.

A Messaging Server knows a participant has lost connectivity if it does not receive a SIP BYE request from that user, but it notices TCP or MSRP connectivity issues with that participant, or receives a SIP BYE request from client side indicating involuntary departure as specified in section 3.4.4.1.3.2 (e.g. containing a Reason header field set to response code 503 (Service Unavailable), as specified in section 5.2.8.1.2 of [3GPP TS 24.229]).

3.4.4.3.9 Re-joining a Group Chat after temporary disconnection, Group Chat is still ongoing

When a Messaging Server Participating Function detects that one of its users' devices is attempting to rejoin an ongoing Group Chat, it will deliver any stored messages and the notifications pertaining to this user's device, and connect the user's session with the ongoing session that the Messaging Server Participating Function controls on behalf of the user.

The user is made aware of the current list of participants in the Group Chat once he has joined successfully and issues a SUBSCRIBE to the conference state for the Group Chat. All participants with any one of these <status> values: 'connected', 'disconnected' or 'pending' shall be included in the resulting SIP NOTIFY request, not just the ones in 'connected' state.

3.4.4.3.10 Rejoining a Group Chat after temporary disconnection, Group Chat is over

When a Messaging Server Participating Function detects that a user's device is attempting to rejoin an ongoing Group Chat which is no longer active, it will forward the rejoin request to the Controlling Function and deliver any stored messages and the notifications pertaining to this user (see Annex B.1.16), and when no Group Chat session was established during forwarding, terminate the session to the client immediately after delivering the last stored message or notification.

As for the case of a 1-to-1 session described in section 3.3.4.1.4, when stored application messages need to be forwarded this will only be done in case the client that accepted supports the application service as indicated in the *a=accept-wrapped-types* SDP attribute and the Contact header field that the client provided in the SIP INVITE request when re-joining the session. Otherwise, according to the operator local policy the messages related to the application shall either remain stored until another client comes online or be converted by the Messaging Server Participating Function into a format that is supported by the client (e.g. a plain text message carrying a link or a descriptive text of the application message's content).

When the receiving RCS Client sends any messages, explicitly leaves the Chat (as described in section 3.4.4.1.3.1) or adds a new participant (as described in section 3.4.4.1.2), the Messaging Server Participating Function shall ensure that the Group Chat is restarted on behalf of the user using the procedure described in section 3.4.4.1.7 and forward the message or action from the user in this session. For notifications sent by the receiving client, the Messaging Server Participating Function shall either send them in a restarted Group Chat session or as SIP MESSAGE requests. In the latter case, these SIP MESSAGE requests shall carry the same RCS Group Chat ID as the Group Chat so that the receiving user can associate them with the Group Chat instead of to a 1-to-1 Chat. The Messaging Server Participating Function should close the delivery session as soon as all stored messages are delivered.

NOTE: When the client subscribes to the conference state event package and the Group Chat is not active (i.e. there is no session between Participating and Controlling Function), as an optimisation this subscription can be accepted by the Messaging Server Participating Function that will consequently generate a SIP NOTIFY request including the last known participant information. In this case the Messaging Server Participating Function shall have to subscribe to the conference state event package when the Group Chat is restarted and link this subscription with the one from the client as a B2BUA. If the client's SIP SUBSCRIBE request is received in a Group Chat that was restarted, it shall be handled as described in section 3.4.4.1.1.

3.4.4.3.11 Storing messages for a participant who has not yet accepted the invitation

The text in section 3.4.4.3.8 applies with the difference that the Messaging Server Participating Function has been storing messages ever since the SIP INVITE request to the participant timed out and it became an endpoint on behalf of that participant by sending a 200 OK.

3.4.4.3.12 Joining and delivering messages for a participant who joins late, Group Chat is still ongoing

When the Messaging Server Participating Function detects that a recipient user's device is now available in IMS and the session with the Controlling Function is still active, stored chat messages and notifications[30] are delivered as described in section 3.4.4.3.13 with the Messaging Server Participating Function's B2BUA linking the session towards the client with the one that was established with the Controlling Function. The Messaging Server Participating Function shall either forward the subscription to conference state request from the client to the Controlling Function after terminating the own subscription or link this subscription from the client to its own subscription as a B2BUA.

3.4.4.3.13 Joining and delivering messages for a participant who joins late, Group Chat is over

When the Messaging Server Participating Function detects that a recipient user's device is now available in IMS, stored chat messages and the notifications[30] targeting this device are delivered as described in section 3.3.4.1.4, with following differences (see Annex B.1.17):

- The P-Asserted-Identity and Contact headers shall include the same values of the corresponding headers in the latest received Group Chat invitation and
- The Referred-By header value shall contain the address of the user who initiated the Group Chat.

NOTE: Whether the Messaging Server Participating Function initiates this action immediately or only after a timeout when it can be assumed that the client will not send an automatic rejoin, is left as an implementation decision. In both cases, the Messaging Server Participating Function should be prepared to handle a race condition between the INVITE request that it sends and a rejoin sent from the client. By delaying the INVITE request, that scenario becomes more unlikely though.

When the receiving RCS Client sends any messages, explicitly leaves the Chat (as described in section 3.4.4.1.3.1) or adds a new participant (as described in section 3.4.4.1.2), the Messaging Server Participating Function shall ensure that the Group Chat is restarted on behalf of the user using the procedure described in section 3.4.4.1.7 and forward the message or action from the user in this session. For notifications sent by the receiving client, the Messaging Server Participating Function shall either send them in a restarted Group Chat session or as SIP MESSAGE requests. In the latter case, these SIP MESSAGE requests shall carry the same RCS Group Chat ID as the Group Chat so that the receiving user can associate them with the Group Chat instead of to a 1-to-1 Chat. The Messaging Server Participating Function shall close the delivery session as soon as all stored messages are delivered. When the user explicitly leaves the session the Messaging Server Participating Function shall next to the forwarding of this action also discard any remaining stored messages.

Forwarding of application messages shall be done as specified in section 3.4.4.3.10. If the first message to be delivered in the Chat is an application message (e.g. a File Transfer via HTTP XML body), the messaging server shall include an Accept-Contact header field in the SIP INVITE request carrying the IARI associated to the application service along with *require* and *explicit* parameters.

---

[30] The Messaging Server Participating Function has been storing messages even if the participant had not joined before.

3.4.4.3.14 Inactive session from Controlling Function activated while forwarding messages

In case the delivery started without an active session from the Controlling Function and the Group Chat is activated from the same Controlling Function as before (i.e. the session identity matches the one provided to the client), the Messaging Server Participating Function shall accept the session from the Controlling Function and connect it as a B2BUA to the one already established with the client. Furthermore, the Messaging Server Participating Function shall on behalf of the user subscribe to the conference state provided by the Controlling Function and connect this subscription as a B2BUA with the subscription from the client that it terminated.

3.4.5 NNI and IOT considerations

3.4.5.1 SIMPLE IM Group Chat – CPM Group Chat interworking

Interworking a participant with SIMPLE IM Group Chat towards a CPM Group Chat server, and a participant with CPM Group Chat towards a SIMPLE IM Group Chat server is done as per the [RCS-CPM-CONVFUNC-ENDORS], via mapping of the appropriate Chat session feature tags when it is determined that the remote network requires such interworking.

3.4.5.2 Messaging Server handling of delivery notifications when not supported by device or terminating network

For devices or networks (e.g. older RCS versions) that the controlling or participating function of an RCS Messaging Server recognises do not support generation of delivery notifications within a Group Chat, the Messaging Server shall generate them on behalf of the devices if such notifications are requested. As it is related to the deployment environment, the method by which the Messaging Server determines whether a device can generate delivery notifications is considered outside of the scope of this specification. The following may be reliable indications though:

- The device indicates explicitly support for the *message/imdn+xml* Mime type in the *a=accept-wrapped-types* SDP attribute.
- The device indicates support for File Transfer via HTTP in the Group Chat (see section 3.5.4.8).

A device that does not generate delivery or display notifications should not receive the CPIM IMDN header in an MSRP SEND that is used to request notifications. When sending a message to a device or network that is assumed not to support disposition notifications in Group Chat, the Messaging Server shall remove this header.

3.4.5.3 Connection Model for Subscriptions from Participating Function to Controlling Function

The Messaging Server Participating Function acts as a subscriber to the conference event package of a Group Chat controlled by the Messaging Server Controlling Function. During an active subscription the Messaging Server Controlling Function notifies the Participating function about changes of the Group State status and participant list.

Messaging Server Participating Functions and Controlling Function may reside in different Service Provider networks. The Connection Model applied on the interface between the two functions shall minimise the impact on the Controlling Function coming from the

Participating Function Service Provider's network topology and service provisioning. This is achieved by limiting the number of active subscriptions per participant in a given Group Chat to "1" based on the Connection Model below.

If the Participating Function initiates a new subscription for a focus session ID on behalf of a participant the Controlling Function shall accept the request provided that the focus session ID exists and the participant is authorised to subscribe to it, i.e. it is a participant in the Group Chat.

At the time of acceptance, if the Controlling Function has another subscription active for the same focus session identity and participant combination then the older subscription should be terminated as defined in [RFC6665].

The Service Provider of the Participating Function shall ensure that the connection model towards the Controlling Function does not restrict the service provided to its users. For Service Providers offering Group Chat for Multi Devices the application of a Back-to-Back User Agent (B2BUA) for subscriptions in the Messaging Server Participating Function is mandatory.

3.4.6 Implementation guidelines and examples

Please note that the following sections propose an experience which is aimed to be employed as a reference for OEM implementations.

3.4.6.1 Protocol flow diagrams

NOTE: To simplify the use cases including the store and forward function the figures and text do not differentiate between different Service Providers and their message servers. These servers are combined to one generic message server function to focus on the client/server communication aspects.

3.4.6.1.1 Start a Group Chat from the Chat application

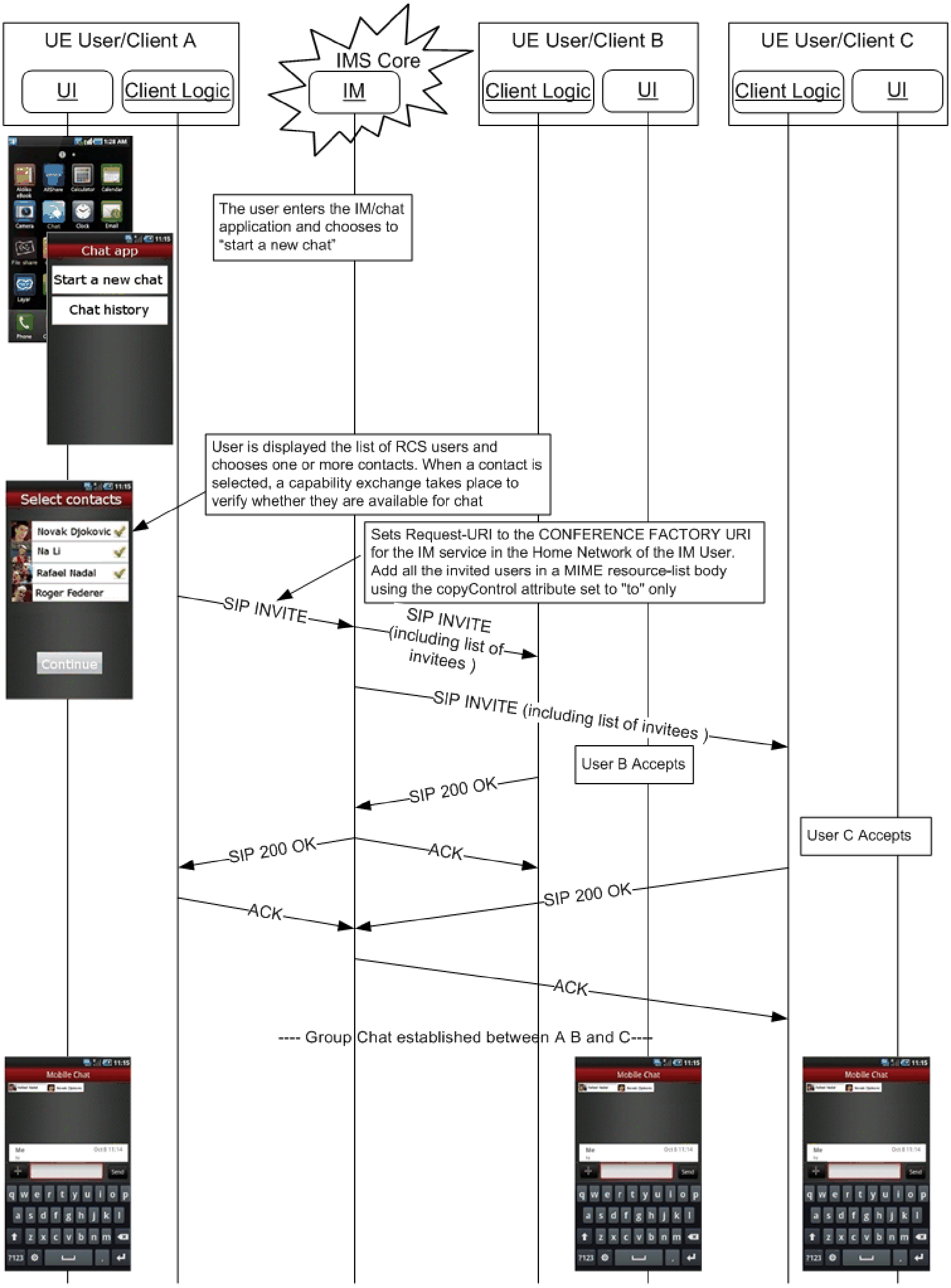

Figure 30: Start a Group Chat from the Chat application

NOTE: The above flow mentions that OPTIONS is used for service capability exchange, which is the case when the DEFAULT_DISCOVERY_MECHANISM is set to 'OPTIONS'. When it is set to 'Presence', then Presence is used.

The UX associated with an RCS Group Chat should provide the following functionality:

- Displaying the list of participants of the current Group Chat and providing of notifications when a new participant is joining and when a participant is leaving the current Group Chat
- When starting a Group Chat session, the invitation shown to the invited users should list the participants invited to the Group Chat before accepting the invitation (e.g. "You're invited to a group chat with A, B & C" instead of "A is inviting you to a group chat")

3.4.6.1.2 Start a Group Chat from the Chat composition window

In this case, Users A and B are in a chat, and User A decides to add a third user (User C) to the chat session. The relevant UX and flow sequence is presented below:

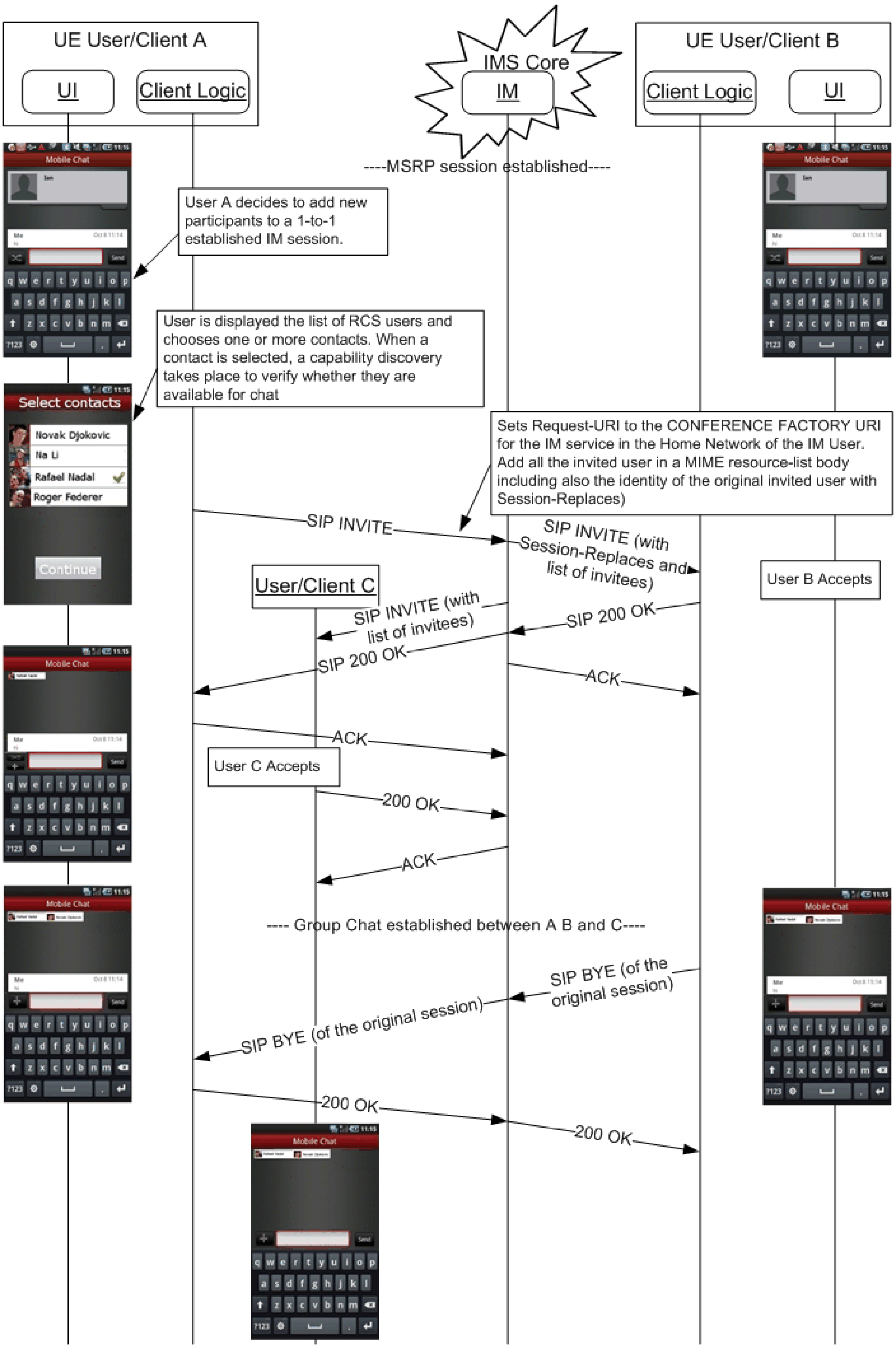

Figure 31: Group Chat session initiation

NOTE: The above flow mentions that OPTIONS is used for service capability exchange, which is the case when the DEFAULT_DISCOVERY_MECHANISM is set to 'OPTIONS'. When it is set to 'Presence', then Presence is used.

3.4.6.1.3 Get participants of Group Chat

The following flow is complementary to the previous use case as it presents in detail how to get information on the chat participants. Please note that these exchanges were omitted in the previous diagram:

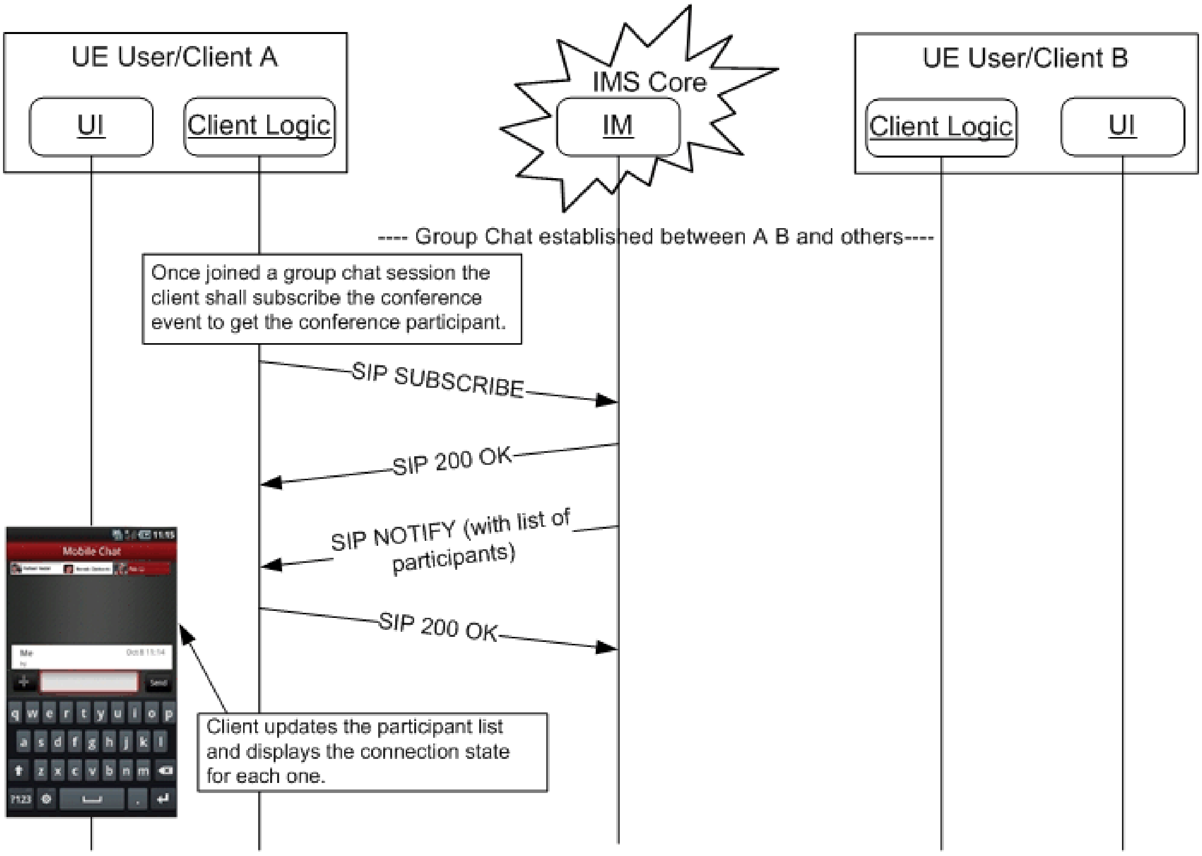

Figure 32: Group Chat session initiation (II): Get participants

3.4.6.1.4 Add a participant to an already established Group Chat

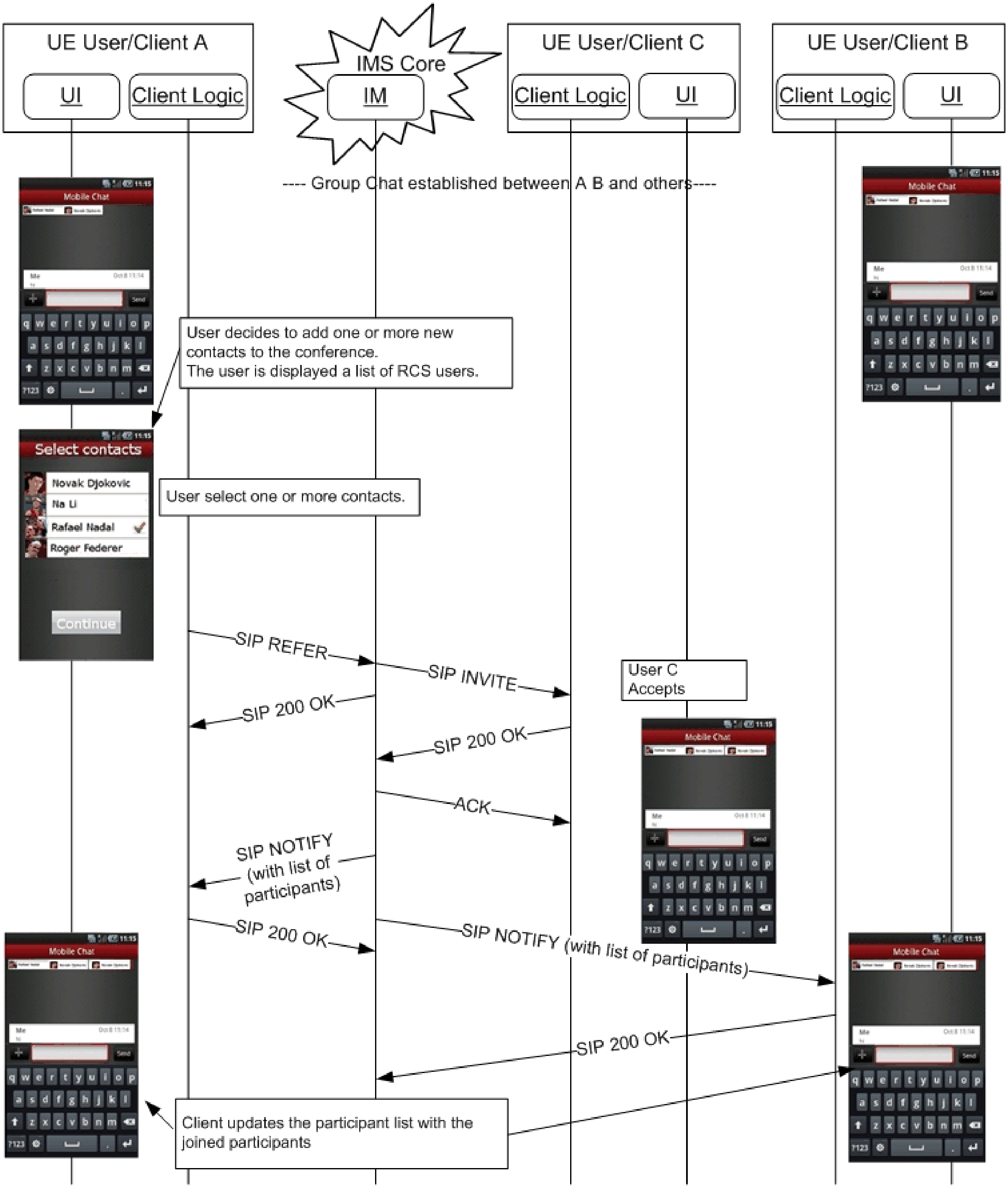

Figure 33: Adding new users to a Group Chat

3.4.6.1.5 Sending a Chat message from the Group Chat window

NOTE: The flow does not show Client B and Client C generating a delivery notification for the received chat message. However, it is expected that they generate one if it was requested.

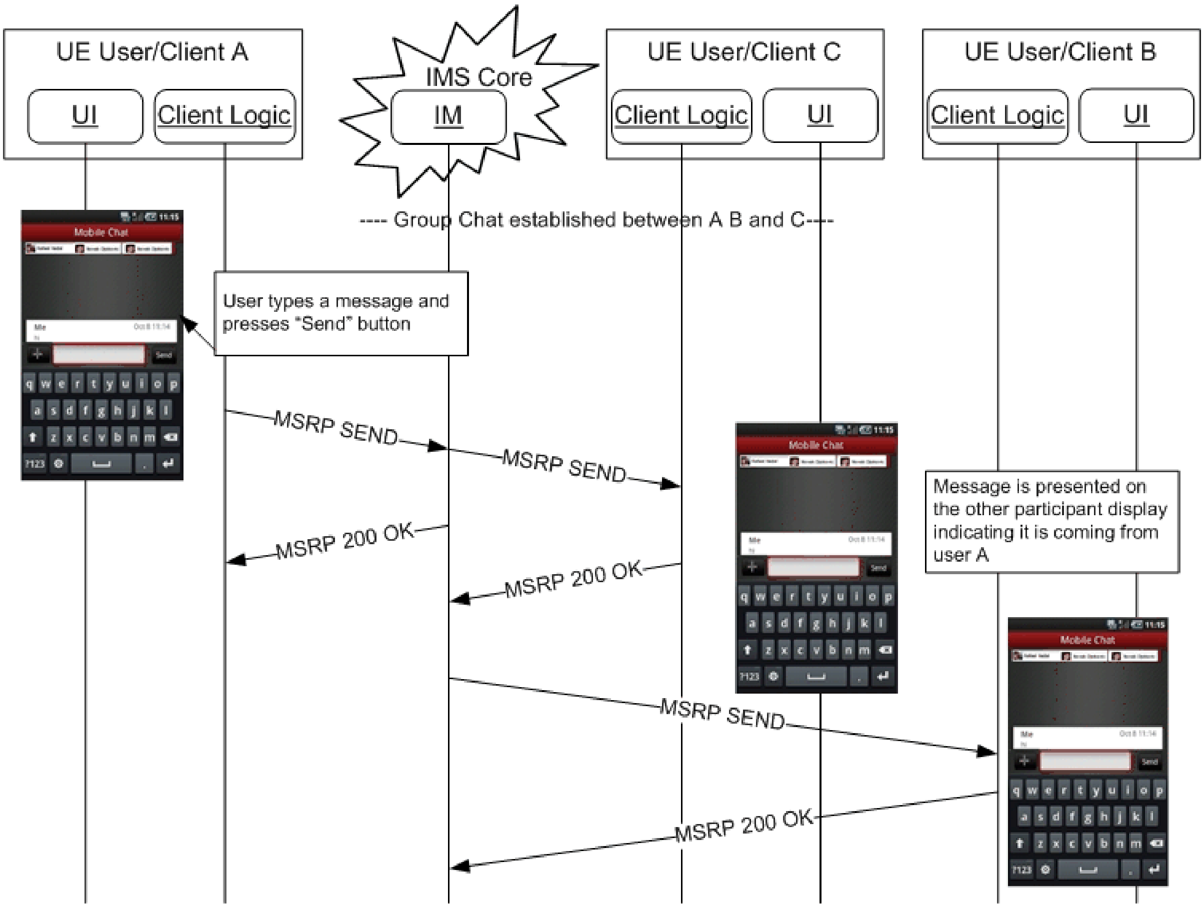

Figure 34: Chat message sequence for a Group Chat

NOTE1: As described in section 3.5.4.8.1, if the message that is exchanged is a File Transfer via HTTP body, the conference focus shall not forward the body to participants that haven't indicated support for File Transfer via HTTP.

NOTE2: As described in section 3.10.4.1.2.2, if the message that is exchanged is a Geolocation PUSH body, the conference focus shall not forward the body to participants that haven't indicated support for Geolocation PUSH.

3.4.6.1.6 Leaving a Group Chat

In this case User B leaves the chat intentionally:

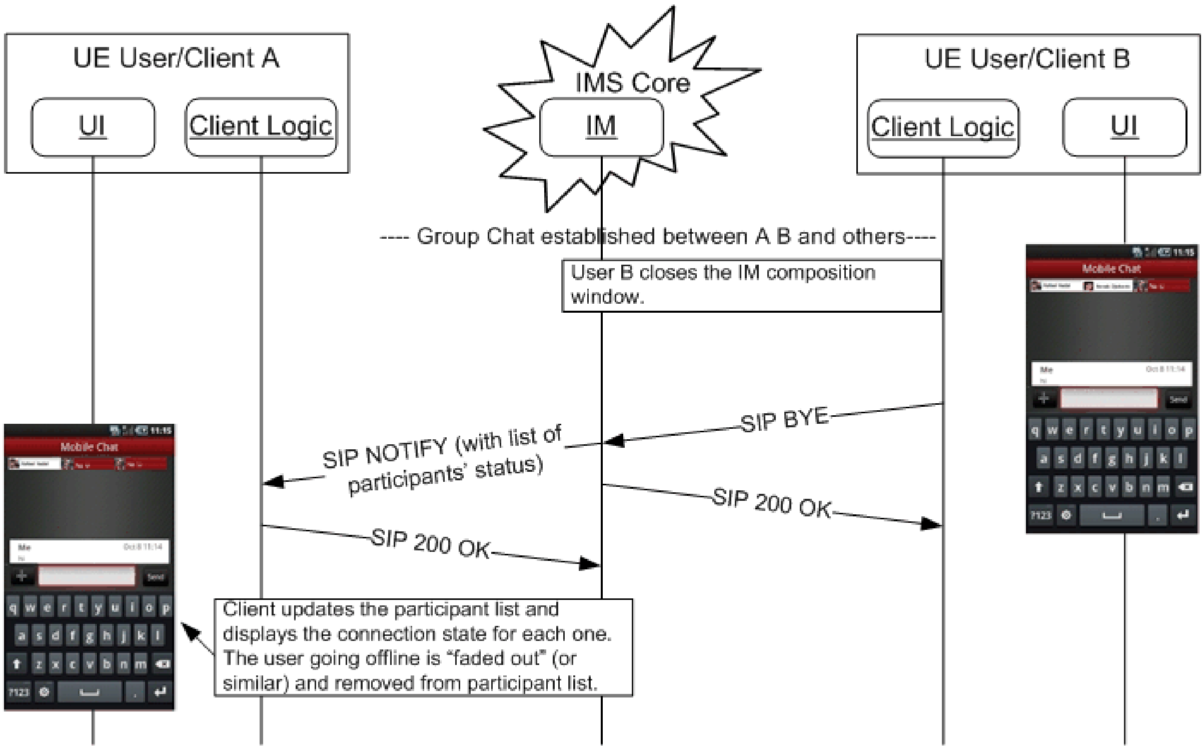

Figure 35: Leaving a Group Chat

3.4.6.1.7 Setting up a Closed Group Chat

In the following flow, User A initiates a Closed Group Chat with Users B and C, but does not invite User D;

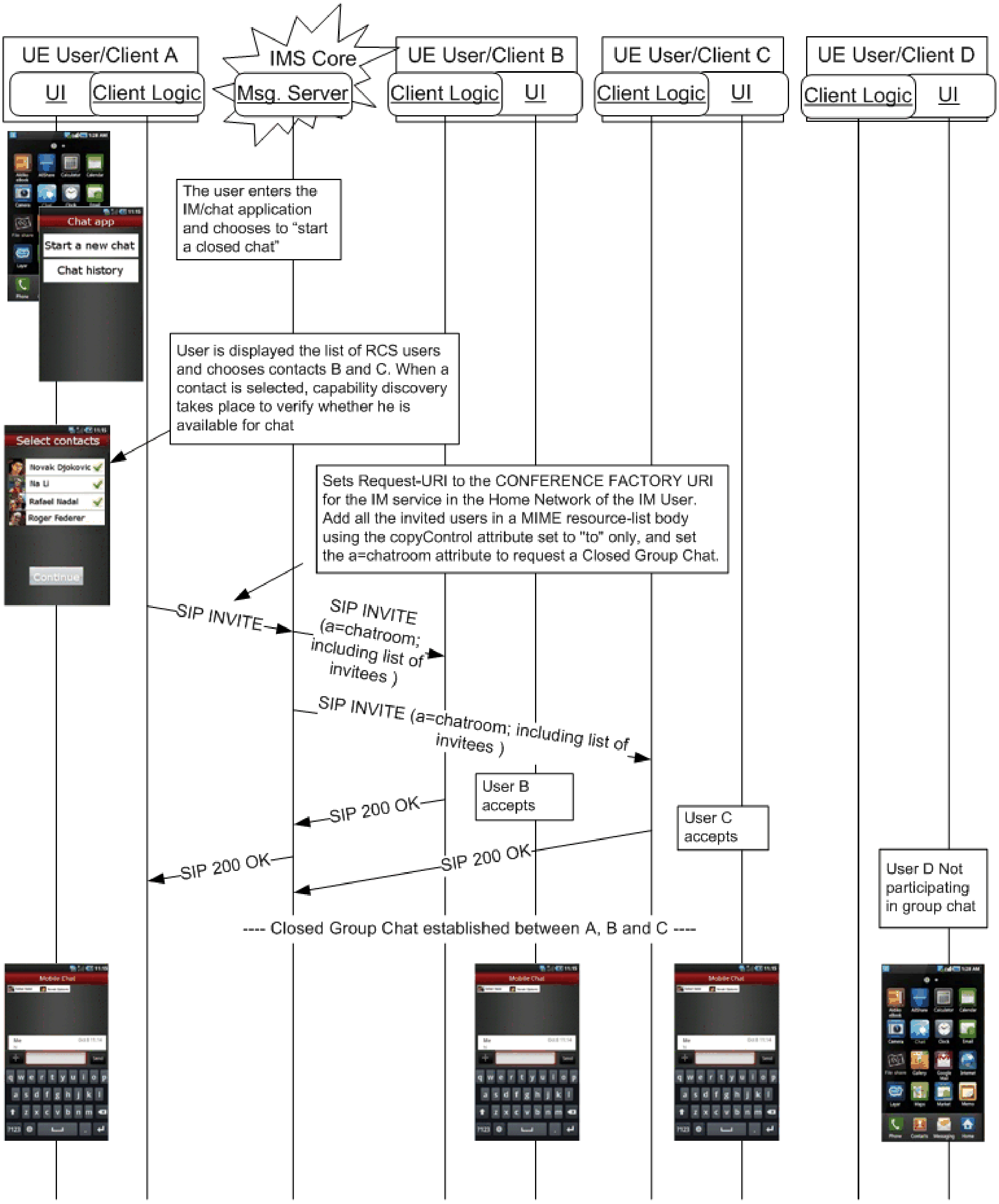

Figure 36: Setting up a Closed Group Chat

3.4.6.1.8 Add new participant for a Closed Group Chat

In the same Closed Group Chat as in Figure 36, User B decides to invite User D, selecting from the list of contacts. Since this is a Closed Group Chat, no additional participants can be added and the Messaging Server will return an error:

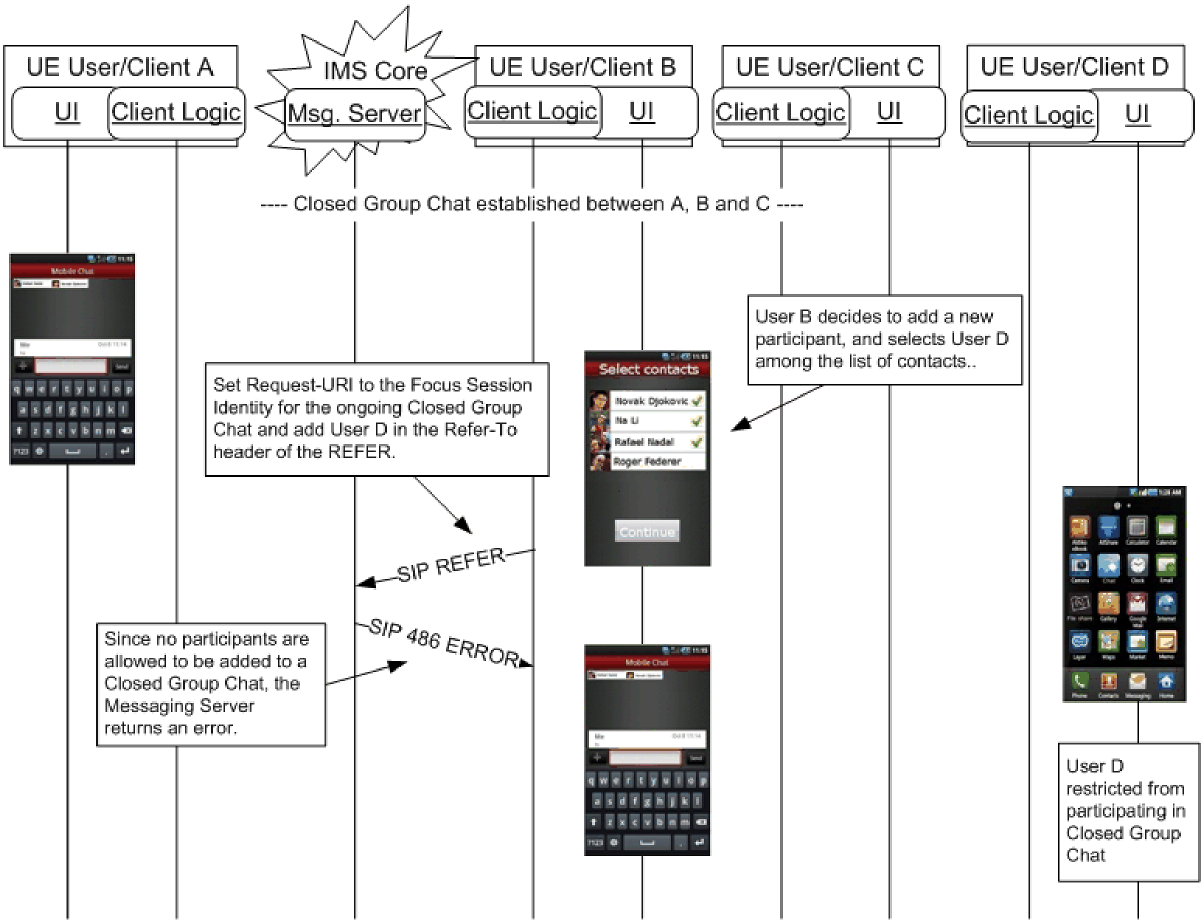

Figure 37: Add new participant for a Closed Group Chat

3.4.6.1.9 Setting up a Group Chat with Store and Forward support

In the following flow, User A initiates a group chat with Users B, C and D. User B subscribes to Store and Forward with Auto-accept capabilities, but decides to join the group chat later. User C manually accepts the group invitation. User D also subscribes to Store and Forward and joins manually.

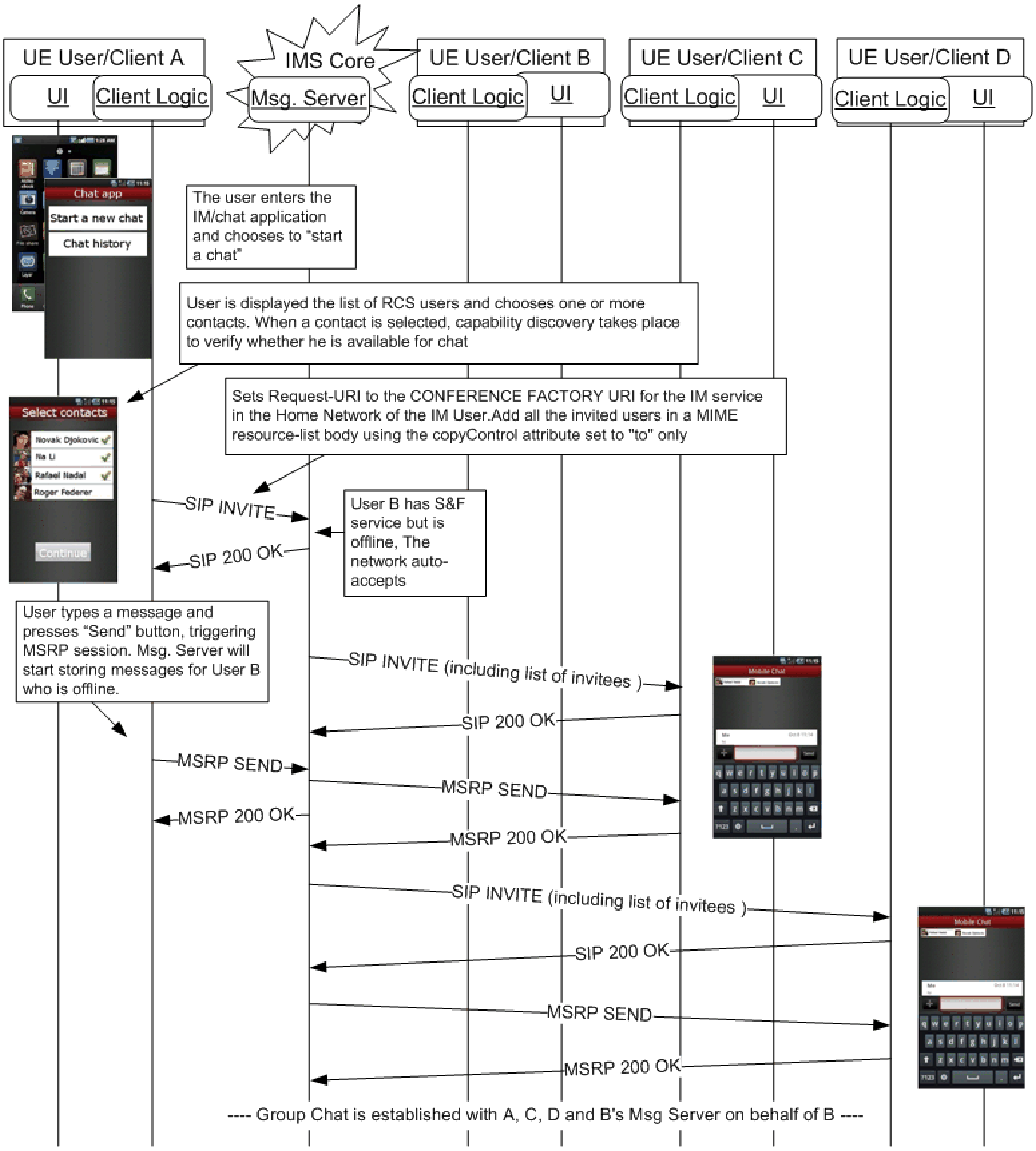

Figure 38: Setting up a Group Chat with Store and Forward support

Figure 39 shows the flow when User B joins the chat:

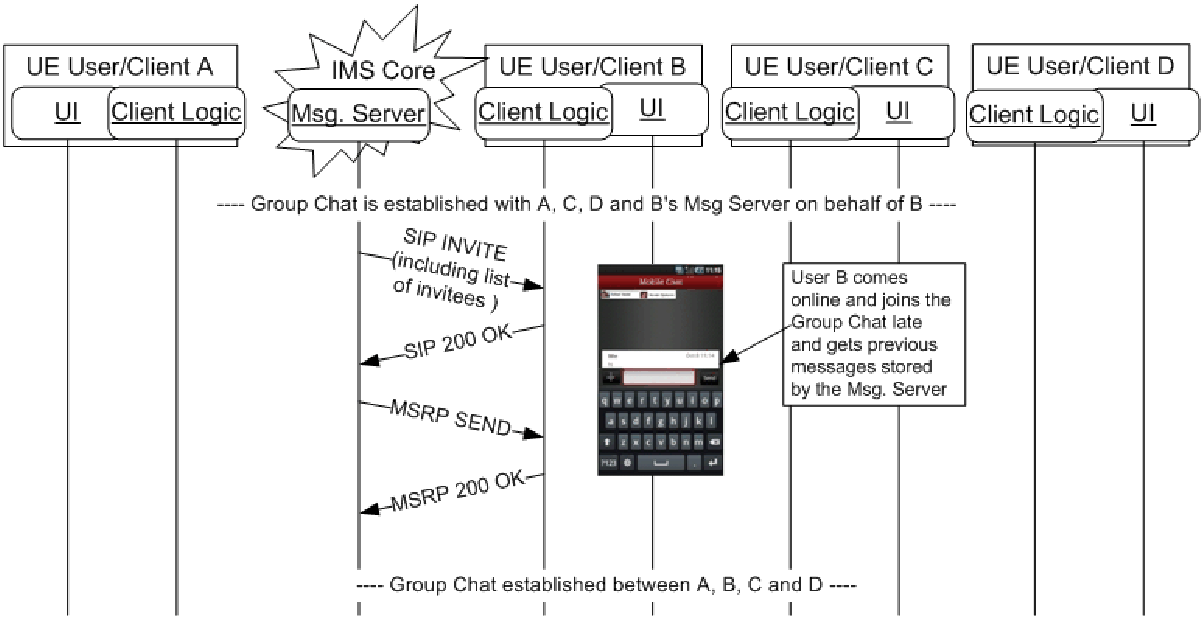

Figure 39: User with Store and Forward joins late

3.4.6.1.10 User in a Group Chat goes offline when Messaging Server supports Store and Forward

In the following flow, Users B and D are in a chat among others (Group Chat); suddenly Users B and D go offline (due to the loss of the connection to the network for example). Messaging Servers supporting Users B and D store subsequent messages received for Users B and D:

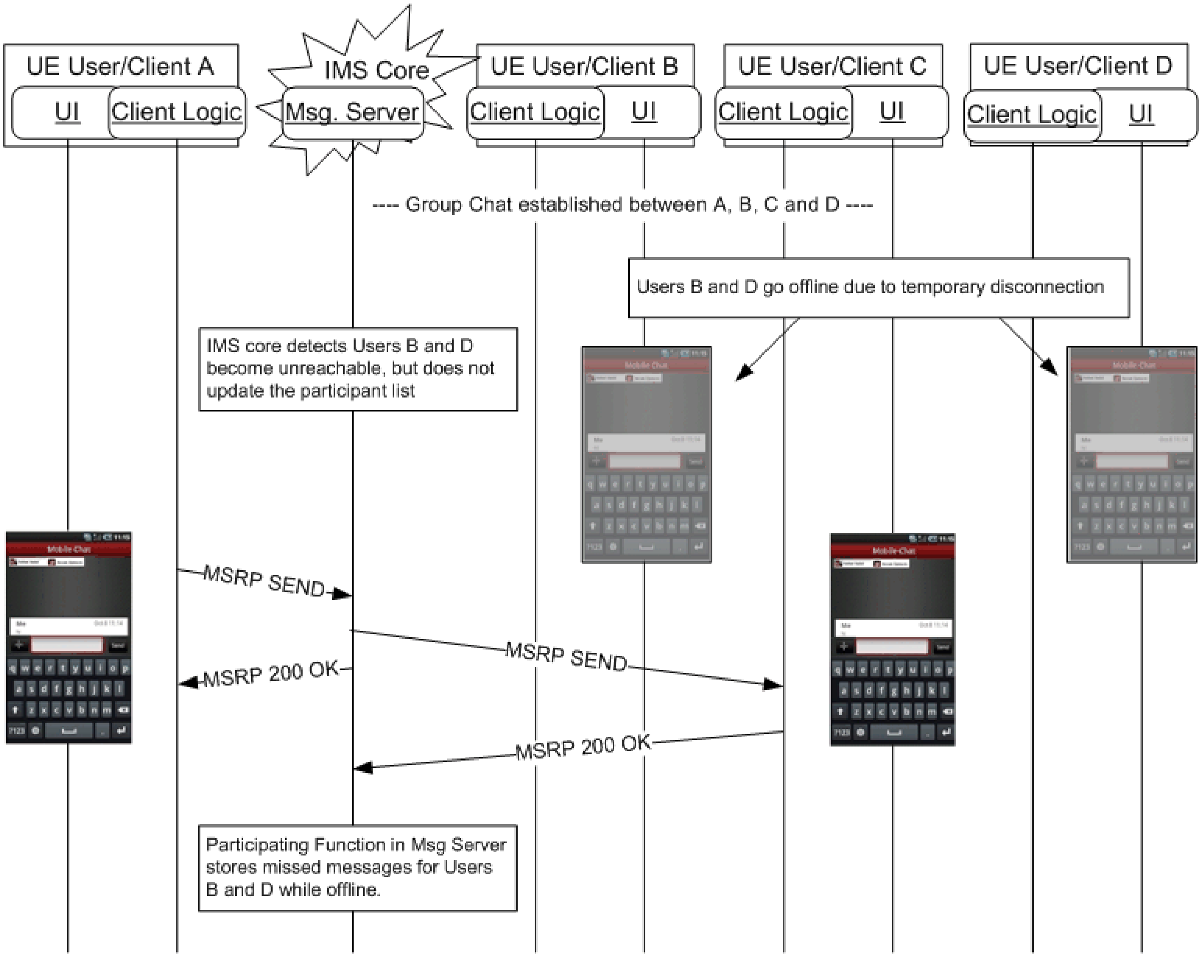

Figure 40: Users go offline when Messaging Server supports Store and Forward

3.4.6.1.11 Re-joining a Group Chat after temporary disconnection, Group Chat is still ongoing

In the same Group Chat as in Figure 40, Users B and D are back online and the Messaging Server will deliver the messages it has stored for both users, and the Group Chat is resumed with all original participants:

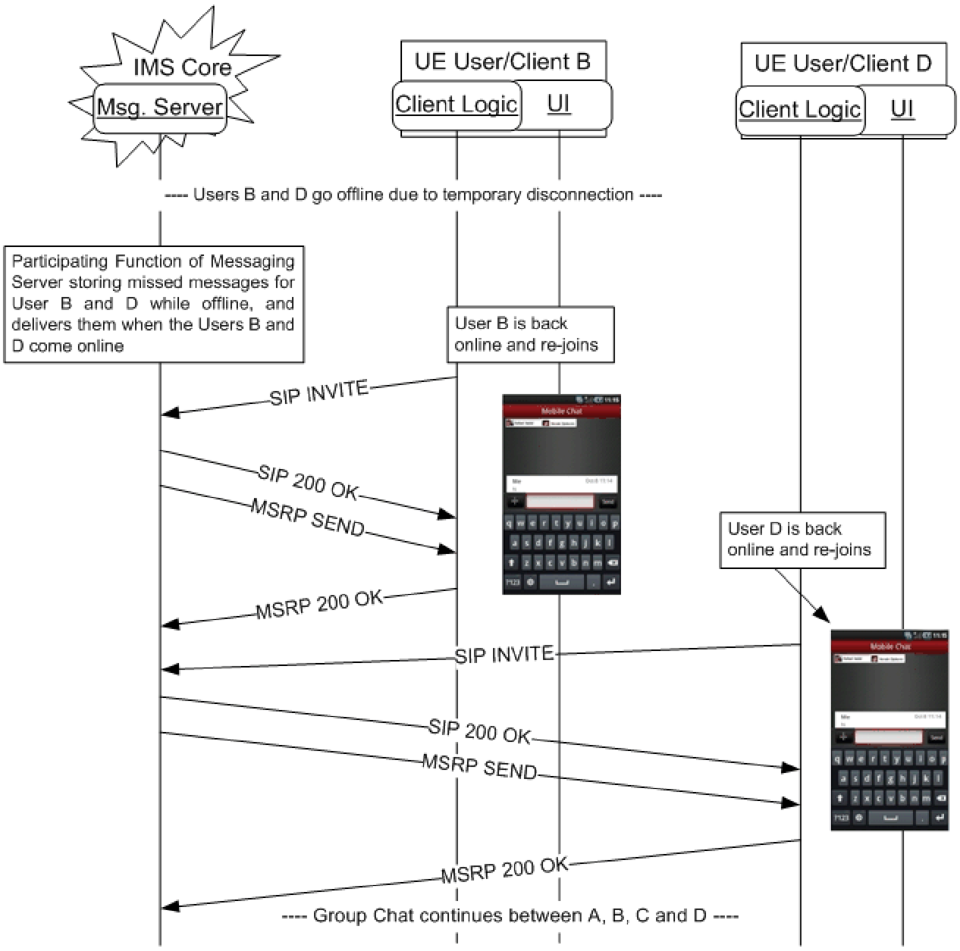

Figure 41: Re-joining a Group Chat after temporary disconnection, Group Chat is still ongoing

3.5 File Transfer

3.5.1 Feature description

File Transfer is the ability for users to exchange different types of content (files), during an ongoing session or without having an ongoing session.

On the sender's side, before sending the request to the intended recipient, the file to be transferred and the recipient have to be selected (refer to use cases in section 3.5.6). If the recipient is not registered, and if the recipient has the File Transfer store and forward service capability, it may still be possible to send the file transfer request (refer to use cases in sections 3.5.4.7 and 3.5.4.7.2.1).

For pictures or video clips, it is a significant added value if the recipient receives a preview of the proposed file before accepting or declining the transfer. Therefore, whenever possible, the sender of the file should include a thumbnail of the file in the File Transfer invitation. A client receiving a File Transfer request with a thumbnail should display the thumbnail in the pop-up presenting the File Transfer invitation.

The request for File Transfer is sent to all of the recipients' devices. When no automatic acceptance is done, this will trigger a pop-up indicating to the user that a contact wishes to send them the depicted file. The recipient is able to select the device to which the file is transferred by accepting or refusing the File Transfer invitation on that device.

In this pop-up shown prior to the actual transfer of a file, the intended recipient is given the opportunity to learn about the proposed File Transfer (size, name, preview and type of file in addition to the identity of the sender) and then to accept or decline the File Transfer based on this information.

If a File Transfer is interrupted for any reason, the receiver can request resumption of the File Transfer without having to re-start from the beginning.

Users are allowed to qualify undesired incoming File Transfer requests as spam. To this end, clients may support a locally stored black list to handle incoming File Transfer requests. This is the same black list as it is used for incoming chat requests. If an invitation to receive a file is received from a blacklisted user, the client should reject the File Transfer request, and from the UX, not notify the user. Instead it may log the event in the spam folder (e.g. “User A tried to send a file on TIME/DATE”).

The File Transfer feature has the following limitations:

- Sharing files with a group of users is only considered within a Group Chat Session. Outside of a Group Chat Session, a device UI may initiate multiple 1-to-1 File Transfer sessions to transfer a file to multiple users**.**
- Only one file can be sent per file transfer session.

3.5.1.1 Handling of specific content

For some of the content exchanged during a file transfer specific handling is provided. This is described in the following subsections

3.5.1.1.1 Card Push

Sharing contact information brings different opportunities to RCS, all of them increasing end user contact possibilities e.g. allowing RCS Users to connect with other RCS or non-RCS Users.

Currently manufacturers are saving the contact info in their address books without following a fully open standard and, as a consequence, sharing this information effectively with other device manufacturers becomes a challenge.

Also, the concept of ‘personal’ and ‘business’ card, representing the user’s own contact information, which may be stored in the address book, is not used simply because this is not an explicit option of the address book menus of existing devices.

This specification aims to:

- Move towards a standard format compliant with all kinds of devices for keeping contact information.
- Create and manage personal and business cards and share them with selected contacts and giving this option visibility in the address book menus.
- Exchange contact information in a secure way.

RCS brings File Transfer as a new service, which becomes a very good bearer for exchanging of contact cards among users. Those contact cards can be sent to another user, like any other file format, using File Transfer.

3.5.1.1.2 Audio Message

The Audio Messaging feature is described in section 3.11.

3.5.2 Interaction with other RCS features

A 1-to-1 File Transfer is not linked to other services (for example, CS-voice call or ongoing chat session) and can be used either during or outside of other communication sessions. The procedure for any file transfer within an ongoing 1-to-1 chat session is implemented as a separate session in parallel with the ongoing 1-to-1 chat and therefore, is the same as the procedure for initiating a separate session for File Transfer.

Different types of content (files) can be exchanged during an ongoing session or without having an ongoing session, i.e. during or outside a call or 1-to-1 chat session.

When transferring a file while not in an existing session (that is when not in a call or chat session with the contact with whom the file is to be shared) and after the transfer has started (that is the receiving user accepted the incoming file) the file transfer is presented to the recipient in a chat context. This establishes a communication context for the transfer since the recipient may want to know why the sender is sharing the file. At the time, the file is presented, the chat session is not started. The chat session will only start if and when the receiver sends a chat message back to the sender of the file transfer.

Figure 42: Reference UX for file transfer on the receiver side

When a file transfer is started during a call with the receiver of the file transfer, the file transfer continues until it is completed or cancelled, i.e. the file transfer will not be terminated when the call ends.

A File Transfer is possible during a group chat. In this case, the file is sent to all participants that are capable of receiving the file.

3.5.3 High Level Requirements

3-5-1 Files can be exchanged during a session (e.g. CS voice call or message conversation).

3-5-2 A File Transfer can be initiated by either end point while having an ongoing session (e.g. the caller or the callee).

3-5-3 End of file transfer shall not lead to termination of a simultaneous ongoing session.

3-5-4 End of a voice call shall not lead to termination of ongoing file transfer.

3-5-5 Files can be exchanged without having an earlier established session (e.g. directly from a multimedia gallery).

3-5-6 The receiver must be able to accept or reject offered files. The invitation procedure shall include an indication to the receiving party concerning file size and type.
3-5-7 The receiver shall have the possibility to save the transferred files.
3-5-8 It shall be possible to assign a service provider configurable maximum file size allowed for File Transfer.
3-5-9 The sending and receiving client shall be able to resume an interrupted file transfer. It is up to Service Provider policy whether only the receiving client or either client can initiate the resume request.
3-5-10 The sending client shall have the possibility to include a thumbnail preview of an offered file.
3-5-11 When sending a file to the recipients in a group chat, the file shall be sent to the network only once.

3-5-12 Store & forward: If the intended recipient is not available, or the recipient does not accept the file transfer invitation within a Service Provider define time limit, the file being offered for transfer shall be available for later retrieval, provided the recipient also has the store & forward service and it is enabled (determined by capabilities exchange).

3.5.4 Technical Realisation

File Transfer is based on [RCS-SIMPLEIM-ENDORS] and [RCS-CPM-CONVFUNC-ENDORS], as well as on the extensions described in [RFC5547]. The technology choice is controlled by the configuration parameter CHAT MESSAGING TECHNOLOGY as described in section A.1.4.3.

A technical variant of File Transfer based on HTTP is also defined in section 3.5.4.8.

SIP INVITE requests for file transfers will be forked to all the recipient's devices. If the recipient accepts the invitation on one device, the corresponding client shall respond with a 200 OK response. If the recipient rejects the session, the client shall respond with a 603 response. In both cases, the IMS network of his Service Provider will cancel the invitations to his other devices.

The SIP 603 Decline response shall be used for the automatic rejection of the incoming File Transfer invitation in case the initiator is included in the device's local blacklist that is described in section 3.5.1.

The current solution provides two complementary technical realisations to provide the file store and forward functionality:

- File fetching/re-delivery via SIP and MSRP: In this implementation, the receiver shall either fetch the stored file using the file transfer fetch procedure described in [RFC5547], or send a new invite to the MSRP server that stored it or wait for a new invitation for the file transfer to be sent by the server that has stored the file previously when the user was offline when the transfer was initiated.
  This is described in sections 3.5.4.2 and 3.5.4.7.
- File fetching via HTTPS: In this technical realization, the receiver shall fetch the deferred file using an HTTPS request.
  This is described in section 3.5.4.8.

The two solutions are complementary; therefore, a service provider can choose the best combination to provide the file store and forward service to their customers maximising the resources that are already deployed in their networks.

The recipient’s client may depending on the setting of client configuration parameter FT AUT ACCEPT (See Table 76 in Section A.1.5) automatically accept the File Transfer invitation from users not included in the device’s local blacklist provided the size indicated in the SDP is below the maximum size for the file transfer warn size (FT WARN SIZE) (See Table 76 in Section A.1.5). Please note in roaming scenarios auto-acceptance shall be disabled by default and if the FT AUT ACCEPT parameter is set, the RCS client shall provide a configuration setting to the user so it can be enabled.

If FT AUT ACCEPT (See Table 76 in Section A.1.5) is set to disable automatic acceptance (i.e. set to a value of 0), File Transfer shall be never auto-accepted.

NOTE: A Service Provider should take into account that enabling any auto-acceptance feature, like the one described above, will impact the multi-device behaviour as it may lead to race conditions.

For File Transfer via MSRP, the extensions described in [RFC5547] are used as follows:

- The SDP payload for File Transfer requests is populated according to [RFC5547], i.e. both sending and receiving clients need to support all elements of [RFC5547]. For populating the file-selector attribute, it is preferable to use the hash-selector, in addition to the other selectors possible. The reason being that the hash-selector uniquely identifies a file, and can also be used to verify the correct transfer of the entire file. The SDP payload shall contain the file size.
- An interrupted file transfer can be resumed by the recipient sending a new SIP INVITE to the originator asking for the missing part of the file. For this it uses the file-range attribute (to denote the missing part) including the file-selector (to denote the file).

NOTE: Absence of the file-range attribute denotes transfer of the entire file.

For such a pull-style operation, the SDP attributes, including file-range and file-selector are populated as described in [RFC5547]. Especially note that from the viewpoint of [RFC5547], this is a new file transfer and hence, it will carry a new file transfer-id attribute.
If the file recipient is required to resume the File Transfer, the SIP INVITE will be forked to all the registered devices of the originator. In that case, any device which has stored the requested file will answer the SIP INVITE with 200 OK if accepted by the user or the RCS client.
For security reasons, an auto-acceptance of resumption requests shall only be offered when a clear correlation between the initial file transfer and the related resumption request can be ensured by the client implementation. In case of manual acceptance, the RCS client application may notify the user that this is a file pull for sake of a resumption request (rather than an ordinary file transfer).

- Generic file pull scenarios (as described in [RFC5547]), i.e. scenarios that do not pursue on a preceding file transfer as described above, are not supported in this specification.
- In scenarios where the file sender notices that an initiated file transfer could not complete successfully, such an interrupted file transfer can also be resumed by the file sending client.

  - The procedure for resumption by the file sending client corresponds to the resumption by the file receiving client described above except for the following differences:

    - The file sending client will send a new SIP INVITE request with a file selector and a proposed file range in the SDP based on information the file sending client has upon detection of the failure condition.
    - The file receiving client will use the file selector and the file range attribute to determine it is a resume request (for this the receiving client may keep information of interrupted file transfers). The file receiving client should include the exact file range required in the SDP returned in the 200 OK on the SIP INVITE request initiating the resumption.
    - Upon reception of the SDP in the 200 OK, the file sending client shall always use the file range specified by the file receiving client for the resume operation.

  - If the file receiving client does not support resumption, the SIP INVITE for the resume will be rejected. The file sending client that initiates the resumption should not continue the resumption. Alternatively, it might then re-send the entire file.
  - If both clients initiate resume, the file recipient's request should be given preference since the file recipient has accurate information about the missing parts of the file. This means that in that case the file recipient client will decline the SIP INVITE request issued by the file sending client.

- If the contact has indicated the capability for receiving a thumbnail through either the inclusion of the *urn%3Aurn-7%3A3gpp-application.ims.iari.rcs.ftthumb* in a SIP OPTIONS based exchange or the *org.openmobilealliance:File-Transfer-thumb* service description in case of a Presence based capability exchange, a preview of an offered file can be added to the SDP description of the SIP INVITE request by using the file-icon attribute of [RFC5547]. The size of this thumbnail shall be smaller than 10 kB. Other SDP attributes will be populated as described in [RFC5547].

  - The procedure describing how to create the thumbnail itself, in its raw binary form, is out of scope of this specification. For a picture, the raw binary result shall be a thumbnail of the picture itself. For a video clip, the raw binary result shall be a thumbnail either of the first I-Frame at 20% of the total length of the video clip or of another relevant frame.
  - The size of a thumbnail should be restricted to the minimum number of octets that provide significance.

NOTE: It is no longer possible to disable the thumbnail capability. As described in section 2.6.1, a client implementing this version of RCS shall always indicate the capability when File Transfer via MSRP is enabled.

In the following sections, the relevant message flows and reference UX are shown. These are based upon the following assumptions:

- For simplicity, the internal mobile network interactions are omitted in the diagrams that are shown.
- It is assumed that by the time the file transfer begins, both the sender and the recipient have exchanged their capabilities using an OPTIONS or Presence exchange. Note that if there is a UX flow that does not show this, the assumption is that the OPTIONS or Presence requests were exchanged between the sender and the receiver (bidirectional) prior to starting the flow.

For File Transfer via MSRP, the RCS client shall populate the P-Preferred-Service header field in all CPM or SIMPLE IM requests with the CPM Feature tag defined for the service, as described in [RCS-CPM-CONVFUNC-ENDORS]. The S-CSCF or AS that performs the service assertion in the originating network shall add the P-Asserted-Service header field set to the value of the asserted CPM service ICSI (i.e. *urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer* for CPM File Transfer, *urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer.group* for CPM File Transfer to a group or *urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred* for deferred delivery done as part of the store and forward realization) and remove the P-Preferred-Service header field before further routing the request.

NOTE: During a transition period towards full compliance, the network support for asserting the service is recommended but not mandatory.

A receiving network element and RCS client should ignore any SIP header fields that they do not understand (e.g. P-Preferred-Service or P-Asserted-Service header fields).

3.5.4.1 File Transfer outside Group Chat

3.5.4.1.1 Selecting the file transfer recipient(s)

The user willing to share a file from the media gallery or file browser will select the file and choose the user with whom the file will be shared. The list that is presented initially to the user may contain RCS contacts not currently registered and to which no store and forward is available. In addition, the capabilities the client has for a contact may not have been updated.

Therefore, the first step is to determine whether the file can be shared with the selected user (that is that user should be registered or be able to receive files using store and forward and the right capabilities should be in place).

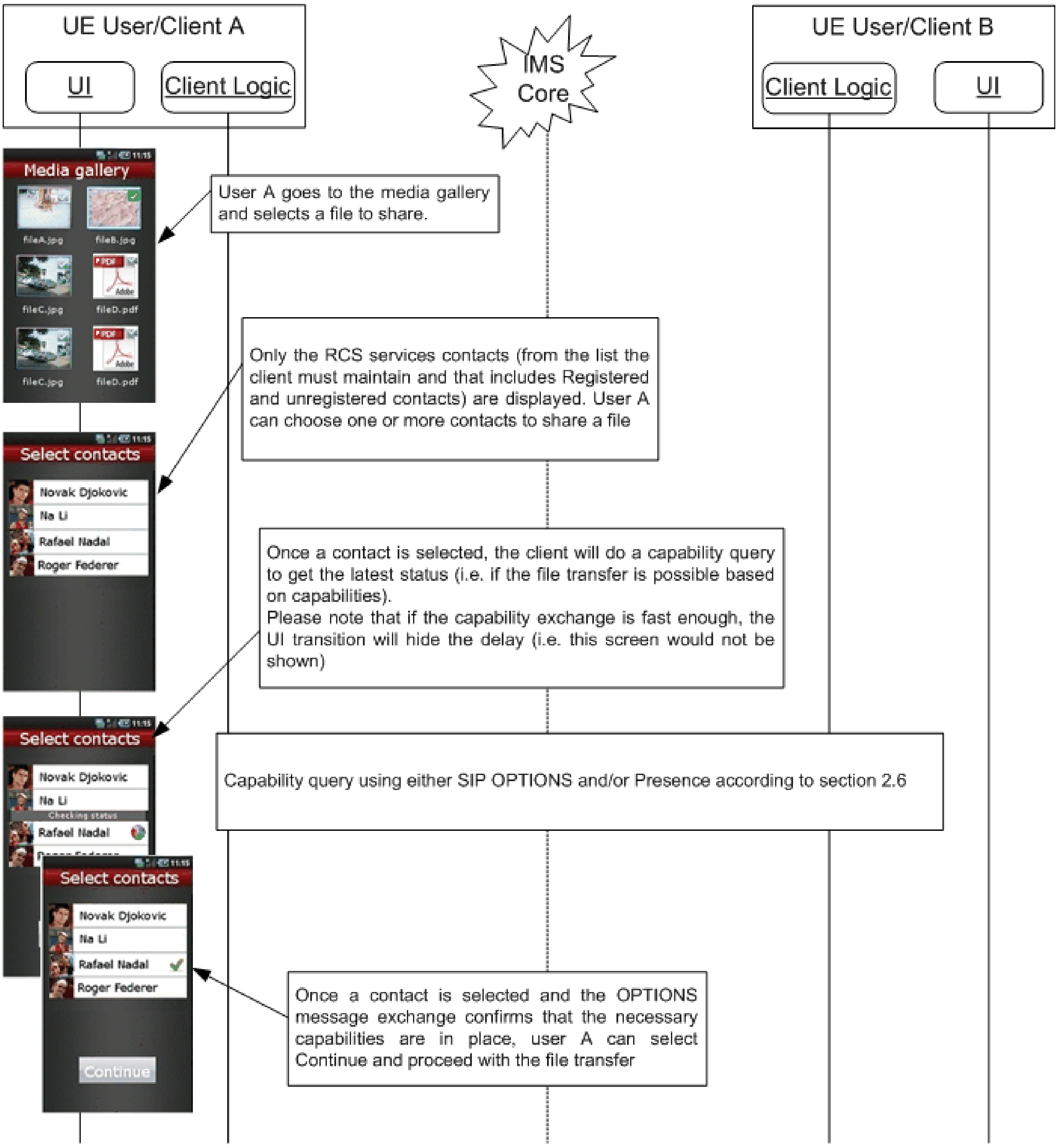

Figure 43: Selecting users when sharing a file from the media gallery/file browser

3.5.4.1.2 Standard file share procedure

Independently of the file share UX entry point, once the file and recipient are selected, the transfer can begin. If a user chooses to share several files, the individual file transfers (in each transfer only a single file is shared) are serialised by waiting for a SIP BYE before issuing the SIP INVITE request for the next file to transfer.

In the following diagram, it is assumed the receiver accepts the transfer.

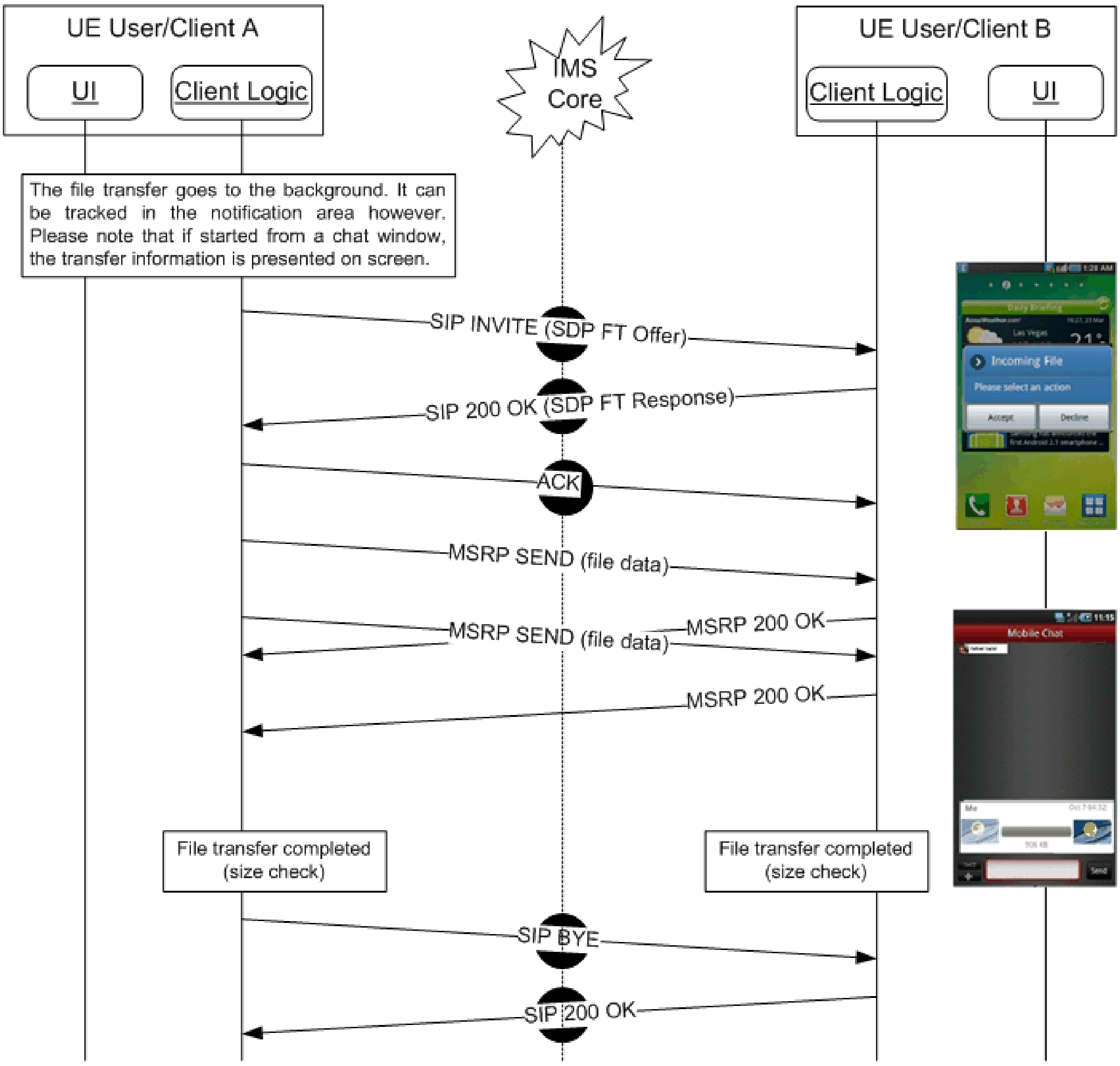

Figure 44: Standard file transfer sequence diagram – Successful transfer

As shown in Figure 44, a client shall send the file in different MSRP chunks without waiting on the MSRP 200 OK response before transmitting the next chunk.

As also shown in Figure 44, for a successful file transfer the client shall only send a SIP BYE after an MSRP 200 OK response has been received to all chunks of the file.

In the following diagram, User B rejects the transfer.

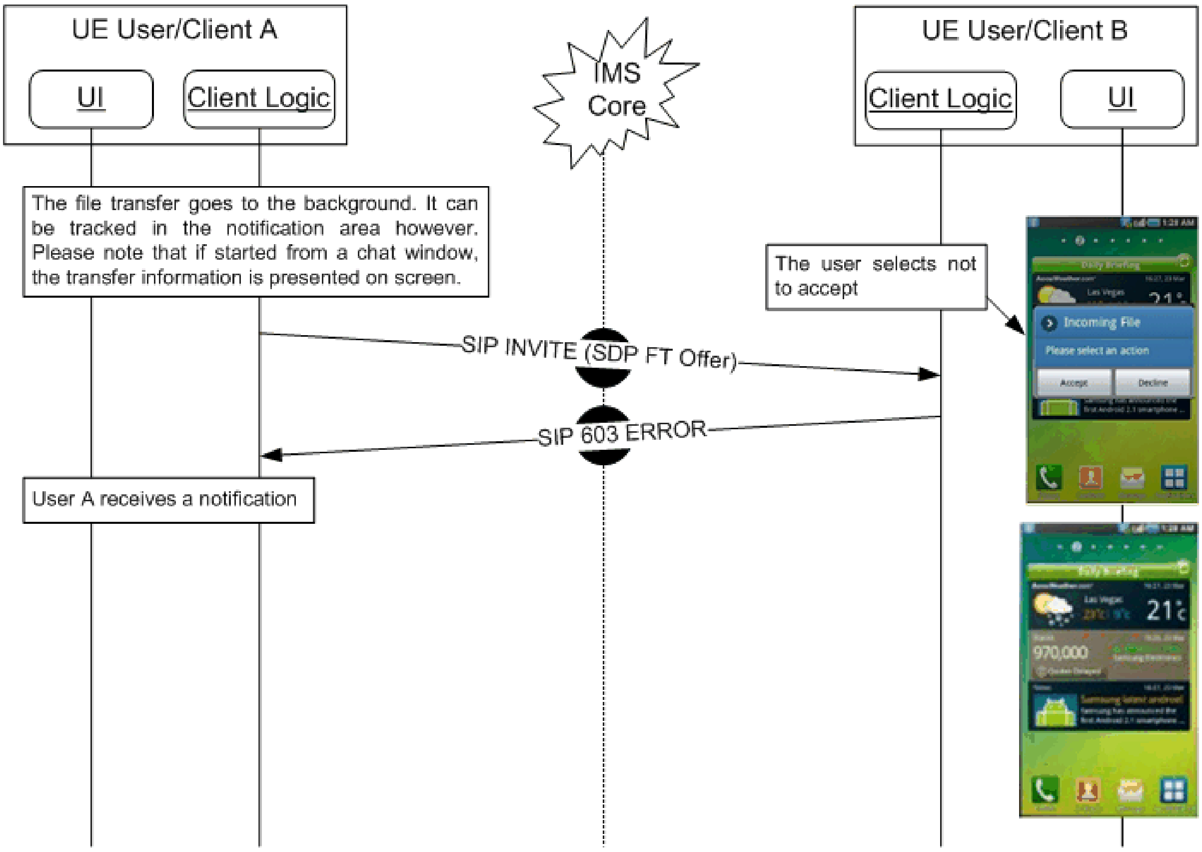

Figure 45: Standard file transfer sequence diagram – Receiver rejects the transfer

3.5.4.2 File Transfer via MSRP in Group Chat

For File Transfer in Group Chat the file shall be sent to the conference focus. To support this the conference focus shall indicate in the Contact header field during the setup of the Group Chat whether File Transfer can be used in the Group Chat by including the IARI tags for the RCS File Transfer services it supports (see Table 7 and Table 11):

- The File Transfer Service itself (including the support for the Thumbnail)
- File Transfer Store and Forward
- Geolocation PUSH (see section 3.10)

NOTE1: These shall be included next to the ICSI for CPM Session Mode messaging or the *+g.oma.sip-im* feature tag if the chat is based on SIMPLE IM.

Similarly, a client initiating, invited to or joining a Group Chat shall include those same tags in the Contact header it includes in respectively the SIP INVITE and 200 OK response for that Group Chat. To indicate its support for this mechanism, a client which is not capable of File Transfer at all shall include the tag for Chat defined in Table 7. Clients supporting File Transfer may include this attribute as well.

Also, a Messaging Server accepting the Group Chat session on behalf of the user in Group Chat store and forward scenarios shall provide its capabilities in the Contact header allowing it to indicate for example whether for that user File Transfer is possible depending on store and forward for File Transfer being supported. When the Messaging Server takes the Group Chat session over from the user (e.g. when the user loses connectivity) or the user takes over from the Messaging Server (e.g. after regaining connectivity), the newly applicable set of capabilities shall be announced to the conference focus using a session refresh (i.e. a re-INVITE with an updated Contact header) sent by the Participating Function.

When the conference focus indicated support for File Transfer, a client that wants to initiate a File Transfer shall compose a multiparty File Transfer Invitation as described in section 10.1 of [RCS-SIMPLEIM-ENDORS] or section 7.4.1 of [RCS-CPM-CONVFUNC-ENDORS] with following differences:

- The invitation shall be targeted at the IM Session Identity associated to the chat and thus to the conference focus instead of to the conference factory.
- The invitation shall take into account the capabilities of the focus. To avoid backward compatibility problems, the client shall for example not include a thumbnail if not supported by the conference focus.
- No recipient-list shall be included.

NOTE2: When File Transfer is not supported by the focus, a client can still send a file by initiating individual 1-to-1 File Transfer sessions to the chat participants.

The participating functions and IMS will route the request to the focus that will generate individual INVITE requests for the participants as described in section 10.4 of [RCS-SIMPLEIM-ENDORS] or section 9.3 of [RCS-CPM-CONVFUNC-ENDORS] with following differences:

- The conference focus shall generate INVITE requests for all clients that indicated support for File Transfer in the Contact Header during the setup of the Group Chat. In the generated INVITE requests, the conference focus will take into account the capabilities of the contact as indicated in that Contact Header and for example remove a thumbnail if one was included and no support for the thumbnail was indicated by the recipient
- The conference focus shall not generate INVITE requests to any recipients that has indicated not to support File Transfer in Group Chat (i.e. only the IARI for Chat was included in the Contact header for that participant).

NOTE3: The conference focus may, based on local server policy, inform the participants in the chat that have indicated not to support File Transfer through a system message of the fact that a file transfer took place that they could not support.

- The conference focus shall handle participants that have not indicated capabilities in the Contact header during the setup of the Group Chat using one of the following options based on local server policy:
    - Not generate any INVITE requests for File Transfer for that participant (i.e. handle them in the same way as participants that have indicated not to support File Transfer in the Chat).
    - Generate a 1-to-1 INVITE request for a File Transfer on behalf of the initiator of the File Transfer without including a Thumbnail or supporting File Transfer store and Forward. If the recipient does not support File Transfer this may fail.

- For those participants that have announced their capabilities through the Contact header, the conference focus may target the INVITE request at only the device of the participant that is handling the chat session through the mechanisms detailed in section 2.11.2.
- The conference focus shall not include a recipient-list-history body in the generated INVITE requests.
- The conference focus may, based on local server policy, limit the number of ongoing file transfers to a participant.

A client can detect that a File Transfer request coming from the conference focus is associated with a Group Chat in which it is involved this because the File Transfer Contact Header matches the one from the associated Group Chat.

If the Group Chat has been closed due to inactivity when the user wants to send the file to the chat, the Group Chat will be reactivated as stated in section 3.4.4.1.7 before sending the file. After the delay to allow this reestablishment as described in the same section, the File Transfer will be initiated to the Group Chat’s focus.

As in the case of store and forward (See sections 3.5.4.7 and 3.5.4.8) client shall use a CPIM wrapper to request delivery reports if the user wants to be informed about the delivery of the file to the different participants. This wrapper shall be handled by the conference focus for those recipients that do not support store and forward (as indicated in the Contact header provided during the setup of the Group Chat session). The conference focus shall send the content to such a participant without CPIM wrapper and when the file is delivered to that participant the focus may send the corresponding delivery report to the sender either through a SIP MESSAGE (similar to the case where it is generated by the participating function or through an MSRP SEND request in the Chat session.

NOTE: Contrary to the Group Chat itself, for a File Transfer the conference focus has to provide all packets sent in the session to a participant that is late to accept rather than just those that are sent from the moment the recipient joins.

If the initiator of the File Transfer loses connectivity during the transfer, the initiator may attempt a resume of the File Transfer after re-joining the Group Chat. If connectivity is lost by a recipient that recipient may attempt a resume after re-joining the Group Chat (as described for File Transfer Store and Forward in section 3.5.4.7). That resume request will not be relayed by the Group Chat focus to the initiator of the corresponding File Transfer though. Instead the focus shall send a SIP *488 Not Acceptable Here* Error Response.

If the Group Chat is terminated while a File Transfer is ongoing, the Group Chat conference focus may, based on local server policy, interrupt the ongoing File Transfer. Whether this is done can depend on the reason the Group Chat session was terminated.

All this leads to the following flow:

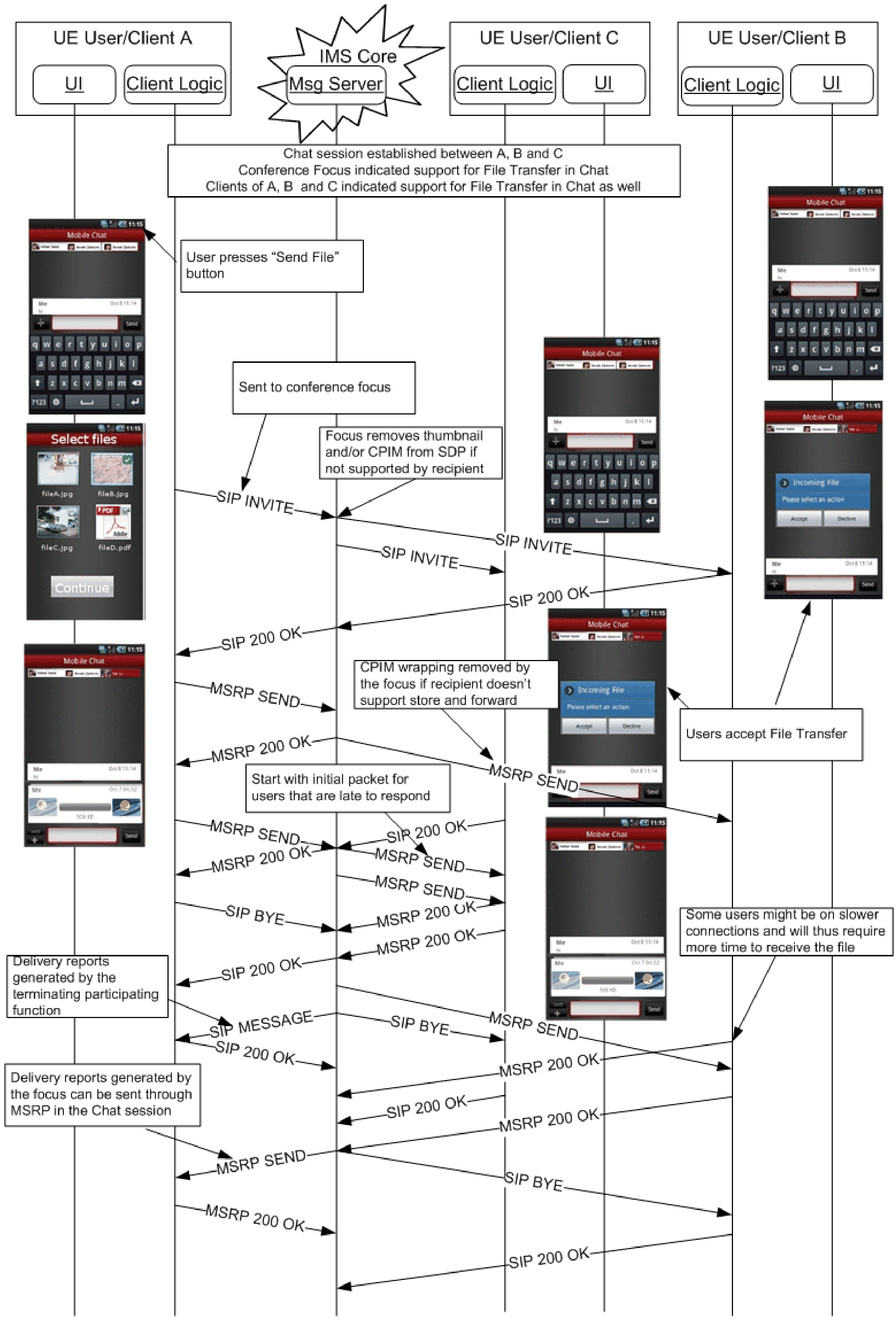

Figure 46: File Transfer in Group Chat

3.5.4.3 File share error cases

There are several scenarios in which a file transfer can result in an error:

Either the sender or the receiver decides to cancel the operation before the transfer is completed. The relevant sequences are equivalent to the diagrams presented for image sharing during a voice call in sections 3.6.4.7.8 and 3.6.4.7.9.

NOTE1: Because in RCS only a single file is transferred in a file transfer session, this simplified procedure for the receiver shall be used instead of the one defined in [RFC5547] (referred to from [RCS-SIMPLEIM-ENDORS] and [RCS-CPM-CONVFUNC-ENDORS]).

Either the sender or the receiver loses the connection to the network before the transfer is completed. The relevant sequences are equivalent to those presented for image sharing during a voice call in sections 3.6.4.7.12 and 3.6.4.7.13.

When transferring larger files, the probability is higher that such a transfer would be interrupted. If such an interrupt leads to termination of the underlying MSRP session, the receiving client, knowing the overall size of the file in transfer, will become a requester of a file (as described in [RFC5547]) and sends a SIP INVITE request, specifying in the SDP payload this file (by using the file-selector as described in [RFC5547]) and the missing part of the file, using the file-range attribute.

Finally, note that if during a file transfer the capabilities of one of the ends change, the file transfer may be affected:

- If the receiver runs out of storage space, the sequence should be equivalent to that presented in section 3.6.4.7.10.
- If on one of the ends a handover into 2G (2G GPRS data coverage) occurs without losing the IP configuration, the file transfer should continue until finished.

If the PNB feature is supported[31], the BPEF checks the recipient of a file transfer against the originating PNB '*rcs_pnb_outft_blockedusers*' of the sender. If the recipient is found on the list, the BPEF will reject the setup of the SIP INVITE session with the blocked user.

NOTE2: For File Transfer, the BPEF may be located in the Messaging Server.

3.5.4.4 File share and file types

In principle the RCS file transfer service comes without a limitation on the file sizes or types. This means that any kind of file can be transferred using this service. Taking this into account and with the aim of providing all the necessary facts to the receiver allowing making an informed decision on whether to accept or to reject the file, a user receiving a file transfer invitation should be informed at a minimum of:

- The size of the file: This is mainly to protect the user from unexpected charges and/or long transfers.

[31] The present section assumes the BPEF as described in section 2.15.1 is provided by the Messaging Server.

NOTE: This also applies to the sender.

- The file type: In this case and to make it more intuitive, the device should present to the user whether the file which is being transferred can be handled/displayed by the device.

For example, if a user receives an invitation to receive a PDF (Portable Document Format) document and their device cannot process that document, an informative message with the size and the fact that the file type is not supported should be presented to the user prior to the user making the decision on accepting or rejecting the file transfer.

Finally note that each individual Service Provider may introduce restrictions taking into account different considerations such as security, intellectual property and so on.

3.5.4.5 Personal Network Blacklists handling

NOTE: In the present section, the BPEF as described in section 2.15.1 may be provided by the Messaging Server.

The Personal Blacklists are applied by the BPEF at both origination and termination of file transfer.

The following *resource-lists* from Shared XDMS are checked by the BPEF by comparing the URI values used in the request and in the list:

- At origination:

  a) The BPEF of the originator checks the '*rcs_pnb_outft_blockedusers*' list to verify that the recipient is not among the blocked users for this request by comparing URIs contained in the list with the URI value of the Request URI of the SIP request.
  b) If this is the case, the message is discarded and a SIP a *403 Forbidden* response with a warning header set to "122 Function not allowed" is returned to the user.

- At termination:

  a) The BPEF checks the '*rcs_pnb_ft_blockedusers*' list, to verify if the originator of the file transfer is among the blacklisted users by comparing the URIs contained in the list with the URI values of the *P-Asserted-Identity* header field of the SIP request.
  b) If true, the BPEF returns a *403 Forbidden* with a warning header set to "122 Function not allowed"
  c) If the Common Message Store is supported it shall store the File Transfer History object data as defined in [RCS-CPM-MSGSTOR-ENDORS] for the blocked File Transfer event in user's dedicated Blocked Folder.

3.5.4.6 File size considerations

To prevent both the abuse of the file transfer functionality and protect customers from unexpected charges, a configurable size limitation (refer to FT WARN SIZE, FT MAX SIZE and FT MAX SIZE INCOMING in Table 76 for reference) may be enabled.

From the user experience perspective and assuming that the size limitation is in place (i.e. the values are non-zero):

- If a file transfer (send or receive) involves a file bigger than FT WARN SIZE, the terminal should warn the user of the potential associated charges and get confirmation from the user to proceed.
- If a file to be sent is bigger than FT MAX SIZE, a warning message is displayed when trying to send and at protocol level following action is taken:
    - For mobile originated file transfer over MSRP the client shall not send the SIP INVITE request.
    - For mobile originated file transfer over HTTP the client shall not upload the file to the HTTP content server.
- If a file is bigger than FT MAX SIZE (when FT MAX SIZE INCOMING is not provisioned) or FT MAX SIZE INCOMING (if it is provisioned) is being received, a warning message is displayed and at protocol level following action is taken:
    - For mobile terminated file transfer over MSRP the client shall send an automatic rejection response SIP 403 Forbidden with a Warning header set to “133 Size exceeded”.
    - For mobile terminated messages the chat message with the file transfer over HTTP file-info element with a file size indication exceeding the limit shall be accepted by the client. The file shall not be downloaded from the HTTP content server. Consequently no display notification will be sent to the other end.

3.5.4.7 File transfer store and forward using a MSRP-based File Transfer Function (FTF)

This functionality requires a logic server function identified as the File Transfer Function (FTF).

NOTE: As a logical function this can be either provided as part of a physical application server that it is already providing analogous functionality (e.g. Messaging AS) or in a separate one.

The client shall assume that the functionality is available when the recipient has indicated the corresponding capability (see section 2.6.1) and the CHAT MESSAGING TECHNOLOGY configuration parameter described in section A.1.4.3 is set to OMA CPM.

This procedure allows the file store and forward mechanism for the following use cases:

1. When the receiver ignores the file transfer invitation causing the SIP INVITE procedure to expire or an early expiration due to an error.
2. When the receiving user is offline.
3. When the either sender or receiver loses connectivity.

This is reflected in Table 32 that shows the error responses that will result in the FTF storing the file:

Response received on terminating leg	Response sent on originating leg	Store the file
480 Temporarily unavailable	200 OK	Y
408 Request Timeout	200 OK	Y
487 Request Terminated	200 OK	Y
500 Server Internal Error	200 OK	Y
503 Service Unavailable	200 OK	Y
504 Server Timeout	200 OK	Y
600 Busy Everywhere	200 OK	Y
Any other response (including 404 Not Found, 603 Decline, 403 Forbidden and 200 OK)	Received response (that is no mapping is done)	N

Table 32: Mapping of received Error Responses by the FTF

3.5.4.7.1 File transfer invitation

If supported by a service provider, this functionality shall be provided by the terminating side (receiver) and, optionally, it can be also provided by the originating side, as per the steps provided below:

1. After the capability exchange takes place, the sending client shall verify that both the sender and the receiver support the file transfer store and forward feature.
2. The original file transfer SIP INVITE shall include a CPIM/IMDN body requesting a delivery notification as described in section 3.3. Note that in this case no message is carried in the CPIM body. This allows the sender to request a delivery notification to confirm when the receiver gets the file.
3. After the SIP INVITE is sent towards the receiver, the terminating FTF shall intercept the message before it is forwarded to the receiver (via normal IMS initial filter criteria already in place for RCS features) and store the file-transfer-id and the file-name SDP attributes defined in [RFC5547].
4. From this moment the terminating FTF shall forward the INVITE to the destination client, and can give a chance for the destination client to accept the File Transfer by waiting for a SIP response during a configured period of time.

    a) If the destination client accepts or refuses the file transfer, before the end of this configured period of time, the standard procedures apply with the precision given in step 5.
    b) Otherwise, after the expiration of the corresponding configured timer, the terminating FTF shall cancel the SIP INVITE request towards the receiver by sending a SIP CANCEL. To make sure the receiver client understands that the reason for this cancellation is a timeout, a reason header shall be included as presented in the following table consistently with [RFC3326]:

Reason: SIP;cause=408;text="User not responding"

Table 33: Reason header in SIP CANCEL due to timeout

    c) If an error occurs that is listed in Table 32, the FTF shall also accept the file transfer on behalf of the destination user and store it.

Please note that specifying when the FTF should accept the initial SIP INVITE and start storing the file transfer is outside the scope of this UNI specification and it is left up to Service Provider policy. Possible implementation choices are:

- Accept the file transfer when the CANCEL is sent to the end user or an error is received (note that this is the option shown in the figures below).
- Accept the file transfer as soon as the initial INVITE has been received.

If the transmission is interrupted from the sender (e.g. because of loss of connectivity), it is left up to the local policy of the FTF whether the received fragment remains stored and if so for what time or whether it is discarded. In case it is discarded (either immediately or after expiry) and the sender tries to resume the file transfer, the file transfer will be rejected as described in [RFC5547]. In that case the sender shall transmit the entire file again as described in section 3.5.4. A stored fragment of a File Transfer shall not be forwarded to the recipient until the sender has resumed the File Transfer and provided the remainder of the file.

5. When the destination client accepts the file transfer invitation by sending a SIP “200 OK”, the terminating FTF should always stay in the media path to be able to have a local copy of the file. How the local copy is performed is outside the scope of this specification and is up to each service provider. If the recipient client loses connectivity, the terminating FTF should complete the File Transfer on the incoming leg anyway in order to be able to, later, provide the file to any potential resume request from the recipient as per procedure defined in section 3.5.4.
The local copy is only needed until the file is received by the recipient. The FTF shall delete the local copy, once the file has been completely received by the recipient.

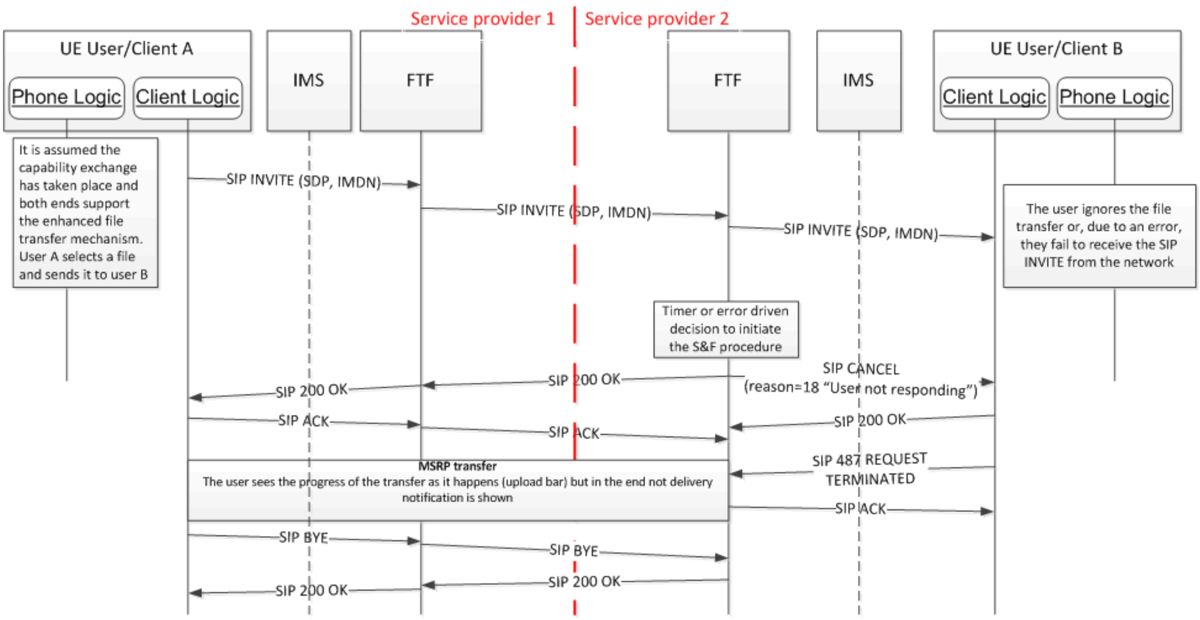

Figure 47: File transfer with store and forward via MSRP fetch on terminating service provider

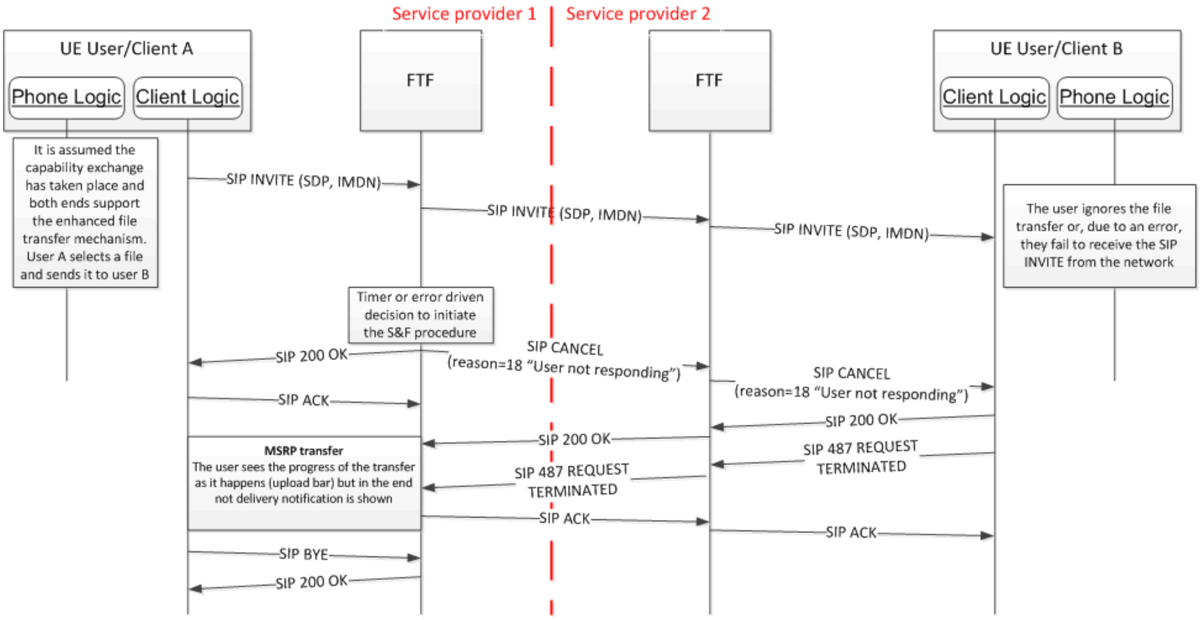

Figure 48: File transfer with store and forward via MSRP fetch on originating service provider

Please note the fetching procedure is covered in section 3.5.4.7.3.

3.5.4.7.2 File transfer to offline users

If supported by a service provider, files can be sent to users that are not online. This functionality can be provided by either the originating (senders) or terminating (receiver's) service provider.

On the originating client this can be enabled by having the *FT CAP ALWAYS ON* parameter (defined in Table 76 in section A.1.5) set to 1, indicating that the file transfer can take place even if the recipient is offline. This parameter should only be set to 1 if either:

1. All the interconnected service providers support the file transfer store and forward feature, or,
2. Store and forward for files is provided as an originating function (sender's FTF).

In this case, File Transfer shall be offered towards all users that are known to support the File Transfer service based on a prior capability exchange.

Also, when *FT CAP ALWAYS ON* is set to 0, the originating client may, based on a prior capability exchange, as described in section 2.6 be aware that the recipient that is offline supports Store and Forward for File Transfer. If *FT CAP ALWAYS ON* is set to 0 File Transfer shall not be offered to offline recipients that are not known to support Store and Forward for File Transfer.

When initiating a File Transfer to an offline user, the client shall compose the file transfer SIP INVITE request as described in [RCS-SIMPLEIM-ENDORS] or [RCS-CPM-CONVFUNC-ENDORS] (depending on the used message technology) with following changes:

1. It shall include a CPIM/IMDN body identical to that described for the chat/IM service in section 3.3.4.1 except that in this case a display notification is not requested and no message is carried.
2. It shall not include the FT thumbnail, since it is not known if the recipient or the recipient's network has this capability.

There are two possible cases:

1. The receiver's Service Provider supports the RCS File Transfer store and forward procedures and is aware that the receiver is offline or receives a SIP 408 Request Timeout or SIP 480 Temporary unavailable error when sending the request to the client, and therefore, accepts the file transfer on their behalf.

   - When the receiver's FTF has detected that the receiver is back online (i.e. third party registration) the FTF forwards the SIP INVITE request without the CPIM/IMDN body. In order to support legacy devices, the file transfer SIP INVITE request shall carry the P-Asserted-Identity of the original sender, rather than the identity of the FTF that stored the message.
   - The receiver's FTF will take the responsibility to issue the delivery notification back to the originator.

2. The sender's Service Provider supports the RCS File Transfer store and forward procedures.

   - In this case, the FTF may not be able to detect when the user comes back online, and must therefore periodically retry to send the File Transfer SIP INVITE request to the recipient. The retry period and duration is determined by local Service Provider policy (see section 3.5.4.7.2.1).

From this point on, the standard file transfer procedure and the cases covered in the remaining sub-sections of section 3.5.4.7 apply.

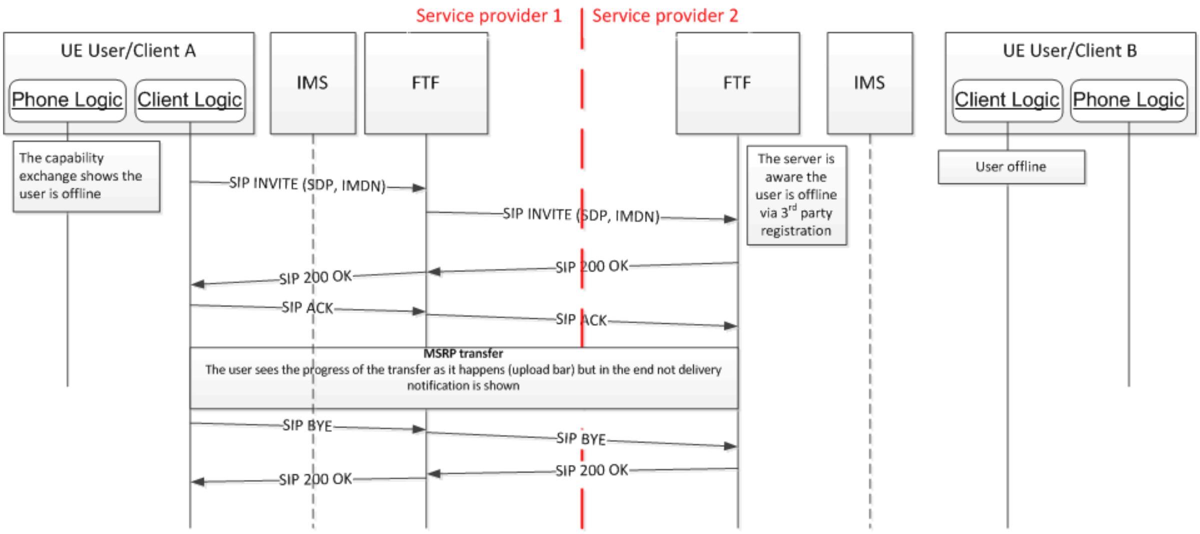

Figure 49: File transfer with store and forward via MSRP for offline users (1/2)

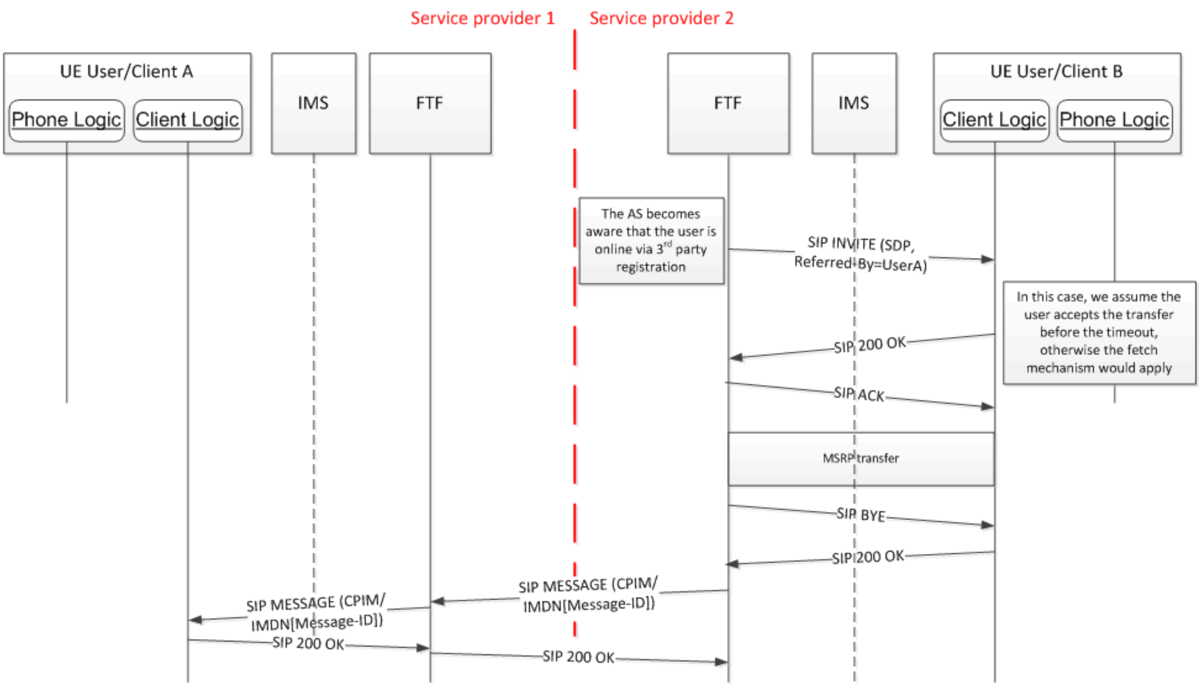

Figure 50: File transfer with store and forward via MSRP for offline users (2/2)

3.5.4.7.2.1 File Transfer retries in originating network

If the sender’s network provides store and forward functionality, the sender’s FTF will accept the File Transfer request if one of error responses listed in Table 32 is returned from the terminating network.

In this case, the originating FTF shall attempt retries to deliver the file towards the receiver with the following procedures:

1. A normal file transfer SIP INVITE request is sent from the sender’s FTF to User B as described in [RCS-SIMPLEIM-ENDORS] or [RCS-CPM-CONVFUNC-ENDORS] (depending on the used message technology) with following changes:
   - The SIP INVITE contains file transfer feature tag and the identity of the original sender in the *P-Asserted-Identity* header.
   - This SIP INVITE shall be sent without the CPIM/IMDN body containing the delivery notification request (i.e. like in the case of a file transfer without the store and forward functionality).
2. When the receiver’s device is online and the user accepts the File Transfer, the file shall be transferred.
3. When the file is delivered, the FTF shall issue the delivery notification back to the originator and should delete the stored copy of the file.
4. If the receiver’s device is not available, a *480 Temporary Unavailable* error can be expected. If that or another error listed in Table 32 occurs:
   - The sender’s FTF re-tries with step 1 after a Service Provider configurable amount of time.

5. If the Service Provider defined number of retries or amount of time has elapsed or a SIP *603 Decline* response is received, the undelivered files are discarded, and the sender is notified if requested.

3.5.4.7.3 Client behaviour and file fetching

After receiving the cancellation (protocol-cause 408 in the Reason header field indicating that store and forward took place), the RCS client shall try to fetch the file as presented below:

1. The receiver client/device implementation, knowing that the original SIP INVITE is expired, shall fetch the file from the FTF. In order to identify the requested file uniquely the client shall:

    a) Use the same SDP file-transfer-id that was used in the original SIP INVITE
    b) Use the same SDP file-name that was used in the original SIP INVITE.

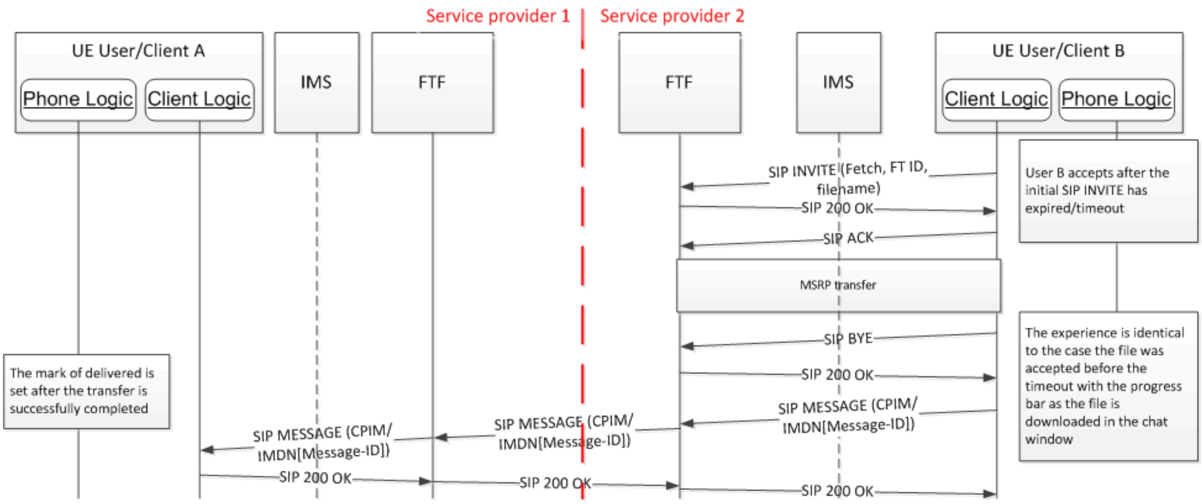

Figure 51: File transfer store and forward via MSRP on terminating FTF fetch

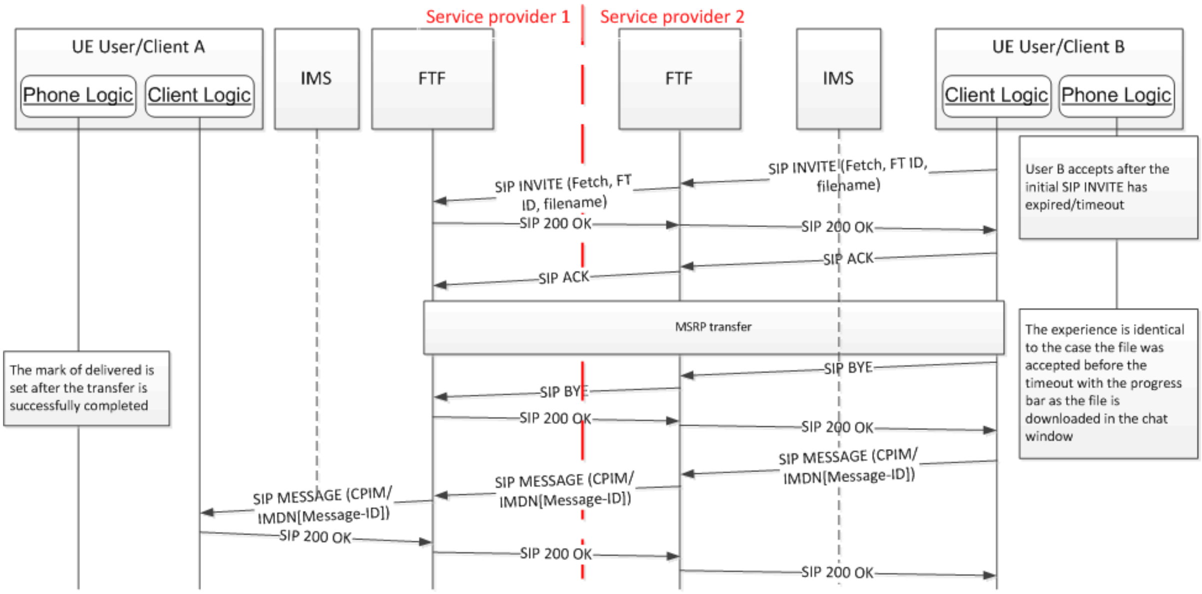

Figure 52: File transfer store and forward via MSRP on originating FTF fetch

2. The server will remove the file once it is successfully downloaded.
3. After the file is successfully downloaded, a SIP MESSAGE containing the delivered notification will be issued to the sender to confirm the destination got the file.

3.5.4.7.4 Timing between originating and terminating store and forward

When implementing store and forward, the timing to trigger the store and forward procedure shall take into account whether store and forward is supported on the terminating side, the timer (or time to trigger the error that signals store and forward is required) shall be significantly smaller than the timer used on the originating store and forward process. Consequently, the following recommendations shall be followed:

- The timer on the originating side should be greater than a half the SIP INVITE timeout period.
- The timer on the terminating side (or time to trigger the error) should be smaller than a quarter the SIP INVITE timeout period.

3.5.4.7.5 File transfer procedures without store and forward

Following the capability exchange and assuming both sender and receiver support the store and forward procedures, there are two possible scenarios where the file transfer procedure does not require a store and forward:

1. If the receiver accepts before the SIP INVITE expiration, then the file transfer takes place as normal:

    a) The MSRP session is established to perform the file transfer.
    b) When completed a SIP BYE is exchanged to terminate the session.
    c) Please note that the original file SIP INVITE is modified to include a CPIM/IMDN body identical to that described for the chat/IM service in section 3.3 except for the fact that in this case a message is not carried and a display notification is never requested. This allows the sender to request a delivery notification to confirm when the receiver gets the file. In this case, as no store and forward takes place, the receiver client is responsible to issue a SIP MESSAGE containing the CPIM/IMDN notification that the file has been successfully delivered.

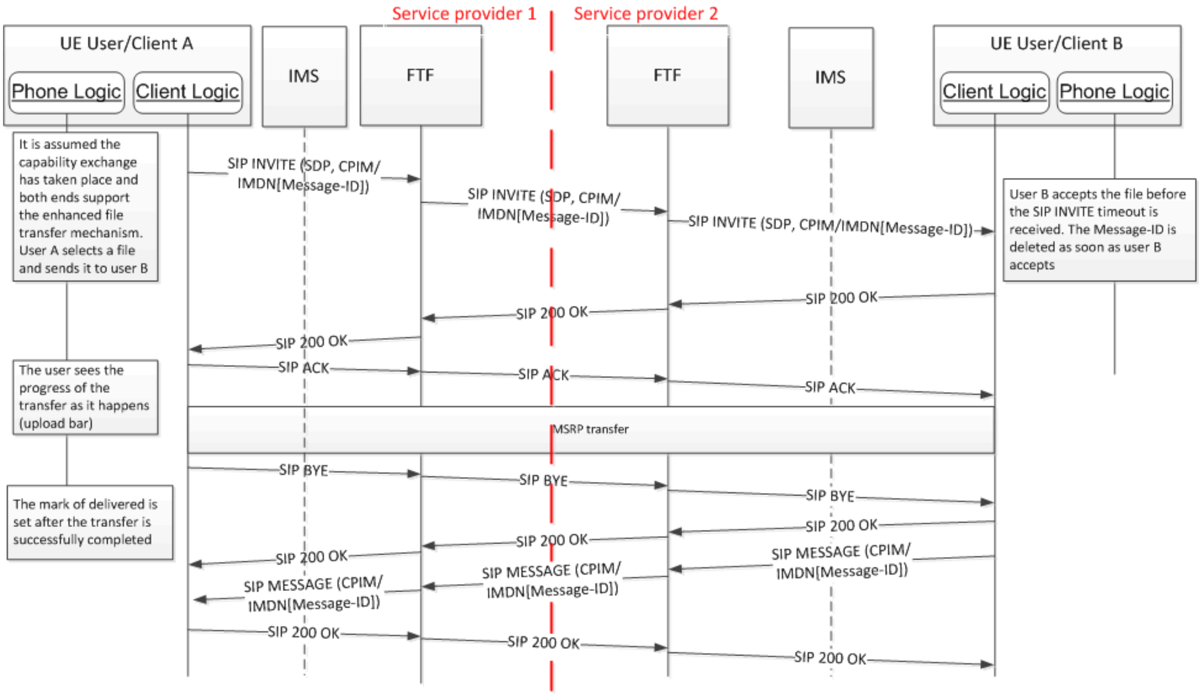

Figure 53: File transfer without store and forward: Receiver accepts file before timeout

2. If the receiver rejects before the SIP INVITE expiration, then a 603 Decline response is sent to the sender and the file transfer is cancelled.

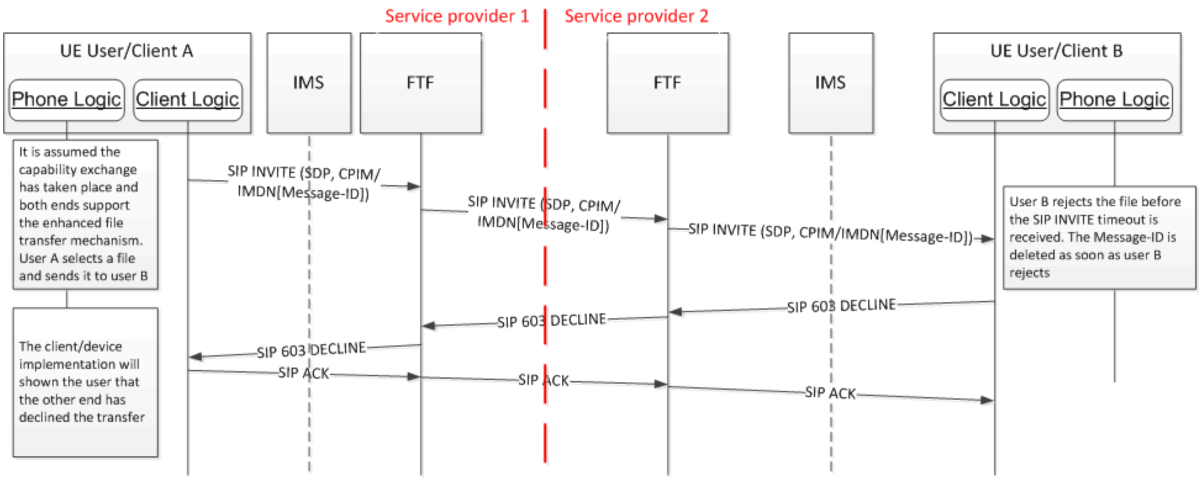

Figure 54: File transfer without store and forward: Receiver rejects file before timeout

3.5.4.7.6 CPIM/IMDN delivery notifications

The same mechanism used for the 1-to-1 chat described in section 3.3.4 specification shall be used with the following changes:

- The notifications shall always be sent using a SIP MESSAGE.
- The IMDN disposition shall ONLY include a delivery notification (request or response depending on the case) and never request or generate a displayed notification.

3.5.4.8 File Transfer via HTTP

As presented in the previous sections, the default mechanism to transfer files in RCS is based in a MSRP transfer.

The present section proposes an alternative mechanism where the file transfer is based in storing the file in a publicly available server and then sharing the location using standalone messaging and the 1-to-1 and Group Chat procedures described in sections 3.2, 3.3 and 3.4. Message revocation procedures as described in section 3.3.4.1.10, do not apply for 1-to-1 Chat messages carrying the location where the file is stored. The same applies for all services that utilise File Transfer via HTTP mechanism (e.g. audio messaging). The main motivations behind this procedure presented below:

- Through the re-use of the procedures for RCS messaging (standalone messaging and chat), the HTTP file transfer mechanism shall automatically benefit from the store and forward mechanism available for chat meaning there is no need to specify additional store and forward procedures.
- HTTP is a quite mature protocol broadly supported for many years in the majority of terminals. This procedure shall, therefore, benefit from its availability and resiliency.

3.5.4.8.1 Configuration and capability exchange

In order to guarantee back compatibility, the file transfer via HTTP procedure shall be only used instead the MSRP procedure if:

1. The sender is adequately configured to use this procedure which is verified by checking that the *FT HTTP CS URI*, *FT HTTP CS USER* and *FT HTTP CS PWD* configuration parameters (all defined in Table 76 in section A.1.5) are correctly set in the configuration received by the file sending client.
2. Both sender and receiver support the procedure by verifying that the File Transfer via HTTP capability defined in section 2.6.1 is present in the RCS capabilities on both ends.

   An RCS client shall only make this capability available if the service is supported by the implementation and the configuration parameters *FT HTTP CS URI*, *FT HTTP CS USER* and *FT HTTP CS PWD* are correctly set. In this case the RCS client shall also include the File Transfer via HTTP IARI tag defined in section 2.6.1.1.2 in the Contact header of the SIP INVITE requests and SIP 200 OK responses that it sends during the setup of a Group Chat.

3. The *FT DEFAULT MECH* configuration parameter (defined in Table 76 in section A.1.5) is set to HTTP.

NOTE: As described in section A.1.5, the *FT HTTP CS URI*, *FT HTTP CS USER* and *FT HTTP CS PWD* configuration parameters could be considered to have been correctly set if they have not been provisioned in case the operator wants to rely on the defaults for those parameters.

If both ends support the procedure, all file transfer shall be performed using the new procedure described in this and the following sections. If not, the file transfer via MSRP procedures described 3.5.4 shall be employed.

When both ends are in chat or group chat session the capability shall be available if following conditions are fulfilled:

- The application/vnd.gsma.rcs-ft-http+xml content type is indicated in the a=accept-wrapped-types attribute during the SDP negotiation and,
- For the case of a 1-to-1 chat, the contact is known to support the File Transfer via HTTP capability based on a capability exchange or on the cached result of an earlier capability exchange when a capability exchange doesn't provide a conclusive result and,
- For the case of a Group Chat, the Contact header field received during the setup of that Chat included the File Transfer via HTTP IARI tag defined in section 2.6.1.1.2.

  A conference focus supporting File Transfer via HTTP shall therefore indicate this support by including the File Transfer via HTTP IARI tag defined in section 2.6.1.1.2 in the Contact Header field of the SIP INVITE and SIP 200 OK responses that it sends during the setup of the Group Chat. When one of the participants in the Chat initiates a File Transfer via HTTP, the conference focus shall not forward the File Transfer via HTTP body to participants that did not include the File Transfer via HTTP IARI tag in the Contact header that they provided during the setup of the Group Chat.

3.5.4.8.2 Offline users

RCS client shall allow the file transfer via HTTP even the receiver is offline when:

- IM CAP ALWAYS ON configuration parameter (defined in sections A.1.4.3) is set to enabled (1), and,
- The receiver user is known to support the file transfer over HTTP capability (cached from the previous exchange).

3.5.4.8.3 File transfer procedure

3.5.4.8.3.1 Sender procedures

NOTE: In this whole section, it is assumed that the sender has the *FT DEFAULT MECH* configuration parameter (see section A.1.5) is set to HTTP.

1. After the capability exchange takes place, it is verified whether both the sender and the receiver support the file transfer via HTTP procedure (as described in section 3.5.4.8.1 and 3.5.4.8.2).
2. Assuming both ends support it, the sender shall first send an empty HTTP POST[32] request (i.e. a request without any body) to the FT HTTP CS URI. If the client supports authentication with an GBA bootstrapped security association as defined in [3GPP TS 33.220] it shall indicate this by addition of a GBA product token in the

[32] This specification uses the term "HTTP POST" and "HTTP GET" as a generic reference to the action of using the POST or GET methods of HTTP. However, it is strongly recommended that whenever the POST action contains sensitive information such as a user ID or password, the action should take place over a secure connection and/or via HTTPS explicitly. This is enforced by the service provider by configuring a FT HTTP CS URI with "https" schema.

User-Agent header as defined in [3GPP TS 24.109]. This request shall result in any of following responses:

a) A HTTP 401 AUTHENTICATION REQUIRED error response carrying a WWW-Authenticate header field as defined in [RFC2616] if authentication is required.
   If the client and the service provider's HTTP content server supports GBA based authentication then the server returns an HTTP 401 AUTHENTICATION REQUIRED response with an WWW Authenticate header instructing the client to use HTTP digest Authentication with a bootstrapped security association. In this case, the client shall authenticate with the bootstrapped security association as defined in [3GPP TS 24.109]. If the client has no bootstrapped security association in place it shall invoke the bootstrapping procedure defined in [3GPP TS 24.109].
   Otherwise, the client and the server shall authenticate with the values of FT HTTP CS USER and FT HTTP CS PWD from the device configuration as defined in Table 76 in section A.1.5.
b) A HTTP 204 NO CONTENT response if authentication is not required
c) A HTTPS 503 INTERNAL ERROR with retry-after header if the server is busy and cannot handle the request. The RCS client shall in this case retry to upload after the time specified in the retry-after header.
d) Any other response, the RCS client shall retry the request in this case.

3. The sender shall then upload the file to the HTTP content server by making a HTTPS POST request to the FT HTTP CS URI to upload the file containing the following elements:

   - A file transfer Transaction ID (TID): this TID is optional and is included in case the client supports the optional resume of the file upload as described in section 3.5.4.8.3.1.1. The TID value shall be a unique ID generated by the client according to [RFC4122] section 4.2;
   - The thumbnail content: This is optional as it is only required for images and videos as per the procedures described in section 3.5.4;
   - The file content.

   In order to carry these elements, the HTTP POST method shall contain a MIME *multipart/form-data* entity body with the following parts that shall be transmitted in the listed order:

   - An optional one containing the transaction ID:

```
Content-Disposition: form-data; name="tid”
Content-Type: text/plain

<Transaction-ID generate by the client>
```

Table 34: First form of the HTTP POST method request to upload the file to the HTTP content server (Transaction ID)

   - An optional one containing the thumbnail:

```
Content-Disposition: form-data; name="Thumbnail"; filename="<local_filename>"
Content-Type: [mime type depending on the thumbnail; e.g. image/jpeg]

<Thumbnail content>
```

Table 35: Second form of the HTTP POST method request to upload the file to the HTTP content server (Thumbnail contents)

- One containing the file:

```
Content-Disposition: form-data; name="File"; filename="<local_filename>"
Content-Type: [mime type depending on the file; e.g. image/jpeg]

<file content>
```

Table 36: Third form of the HTTP POST method request to upload the file to the HTTP content server (file contents)

The client should include the Content-Length header to indicate the size of the HTTP request body, as described in [RFC7230]. If present, the Content-Length indicate the size of HTTP POST body part, i.e. the multipart/form-data entity body.

If the client received a HTTP 204 NO CONTENT response in step 2, then it shall not add an authorization header to the HTTP POST request.

If the client received a HTTP 401 AUTHENTICATION REQUIRED response in step 2, with a WWW-Authenticate response header instructing the client to apply basic or digest authentication, then the client shall add an Authorization header to the HTTP POST request in accordance with the requested authentication scheme as per [RFC2617] using the *FT HTTP CS USER* and *FT HTTP CS PWD* configuration parameters as credentials.

If the client received a HTTP 401 AUTHENTICATION REQUIRED response in step 2, with a WWW-Authenticate response header instructing the client to apply digest authentication with a bootstrapped security association, then the client shall use the stored key material and the B-TID to generate keys specific to the HTTP content server as defined in [3GPP TS 33.220]. The client shall add an Authorization header to the HTTP POST request generated from the key material and the B-TID .

The client shall send the HTTP POST request to the HTTP content server.

4. There are two possible cases:

   a) If the upload is successful, the client shall get a HTTPS 200 OK response containing a XML in the body that specifies:

      i. The URL, size, content type and validity for the thumbnail, if applicable.
      ii. The URL, size, filename, content type and validity for the file.

```
<?xml version="1.0" encoding="UTF-8"?>
< file xmlns="urn:gsma:params:xml:ns:rcs:rcs:fthttp">
        <file-info type="thumbnail">
                <file-size>[thumbnail size in bytes]</file-size>
                <content-type>[MIME-type for thumbnail]</content-type>
                <data url = "[HTTP URL for the thumbnail]" until = "[validity of the thumbnail]"/>
        </file-info>
        <file-info type="file">
                <file-size>[file size in bytes]</file-size>
                <file-name>[original file name]</file-name>
                <content-type>[MIME-type for file]</content-type>
                <data url = "[HTTP URL for the file]" until = "[validity of the file]"/>
        </file-info>
<file>
```

Table 37: HTTP content server response: XML contained in the body

Please note that referring to the XML body in Table 37:

- The thumbnail part is only included if the sender uploaded a thumbnail to the server.
- The validity of the files shall be specified by providing the date the files shall be removed on the server using the [ISO8601] format including the date and time in UTC (Coordinated Universal Time) time zone (e.g. 2007-04-05T14:30:00Z). The validity depends on the configuration the originating Service Provider has set on the HTTP content server.

During the upload process the RCS client shall show the user the progress of the upload as in the case for the file transfer via MSRP.

b) If the upload is not successful, there are two cases to consider:

   i. If the server is busy and cannot handle the request a HTTPS 503 INTERNAL ERROR with retry-after header. The RCS client shall retry to upload after the time specified in the retry-after header.
   ii. If any other error, the RCS client shall automatically retry the upload as described in section 3.5.4.8.3.1.1 up to a maximum of three times.

5. When the upload in step 4 was successful, the sender shall then send a message to the receiver(s) with the following content:

```
<?xml version="1.0" encoding="UTF-8"?>
<file xmlns="urn:gsma:params:xml:ns:rcs:rcs:fthttp">
        <file-info type="thumbnail">
              <file-size>[thumbnail size in bytes]</file-size>
              <content-type>[MIME-type for thumbnail]</content-type>
              <data url = "[HTTP URL for the thumbnail]" until = "[validity of the thumbnail]"/>
        </file-info>
        <file-info type="file" file-disposition="[file-disposition]">
              <file-size>[file size in bytes]</file-size>
              <file-name>[original file name]</file-name>
              <content-type>[MIME-type for file]</content-type>
              <data url = "[HTTP URL for the file]" until = "[validity of the file]"/>
        </file-info>
<file>
```

Table 38: File transfer via HTTP message body content

Where compared to the body received from the content server (i.e. Table 37), an (optional) attribute has been added:

- The *file-disposition* attribute to the file-info element of the main file: This optional attribute provides functionality similar to the File-Disposition SDP attribute in file transfer via MSRP which is described in [RFC5547] and can take the same values (i.e. *render* and *attachment*). If the attribute is not included *attachment* shall be used as the default value.

NOTE: Independently of the mechanism used to transport the message (standalone message or chat), a CPIM body will be used. As the content is now an XML, the CPIM content-type property shall be *application/vnd.gsma.rcs-ft-http+xml*.

If sending to a single user, there are two possible scenarios:

- If there is a 1-2-1 chat session established with the user and File Transfer via HTTP is supported in the session as described in section 3.5.4.8.1, the session shall be reused to convey the content shown in Table 38 in a chat message.
- There is no session established:
    - If the RCS client is configured to use standalone messaging and the recipient supports standalone messaging as well, the mentioned message body shall be delivered using a standalone message carrying a dedicated Accept-Contact header field that includes the File Transfer via HTTP IARI tag defined in section 2.6.1.1.2 along with *require* and *explicit* parameters.
    - If standalone messaging is not supported by at least one of the parties, then a 1-to-1 Chat Session shall be established as specified in section 3.3.4. The RCS client shall include a dedicated Accept-Contact header field that includes the File Transfer via HTTP IARI tag defined in section 2.6.1.1.2 along with *require* and *explicit* parameters in the SIP INVITE request that it generates to establish this Chat session. The XML message shall be relayed in this session as follows:

- If the configuration allows including the initial chat message in the SIP INVITE for a 1-2-1 chat, then it shall be used to carry the message.
- If not, the file shall not be sent until the chat session is established.

NOTE: The inclusion of the Accept-Contact header field is only intended to guarantee that the request is routed to devices capable of File Transfer via HTTP. On the receiver's side this request can be handled as a regular invitation for Chat.

If sending to multiple users, there are two possible scenarios:

- If the file is to be transferred in an existing group chat, the session shall be reused to convey the content described in Table 38 in a chat message. If the Group Chat is closed due to inactivity, it shall be restarted first.
- There is no session established:
  - If the RCS client is configured to use standalone messaging and prior verification that all participants support standalone messaging, the mentioned XML message body shall be delivered using a standalone message with multiple recipients carrying a dedicated Accept-Contact header field that includes the File Transfer via HTTP IARI tag defined in section 2.6.1.1.2 along with *require* and *explicit* parameters.
  - If standalone messaging is not enabled a group chat session shall be established first with all the participants before sending it as a message.

When establishing a Chat Session, clients shall indicate their support for this File Transfer mechanism by including the *application/vnd.gsma.rcs-ft-http+xml* in the *accept-wrapped-types* attribute in the SDP that they provide as body in the SIP INVITE request or 200 OK response they send to take part in the Chat and include the File Transfer via HTTP IARI tag defined in section 2.6.1.1.2 in the Contact header field of that request/response. This will ensure that the conference focus does not forward the body to clients that do not support the mechanism as described in section 3.5.4.8.1.

6. The use in the client UI of the delivery notification coming from the receiver when the chat message containing the XML is delivered is left up to the RCS client implementation.

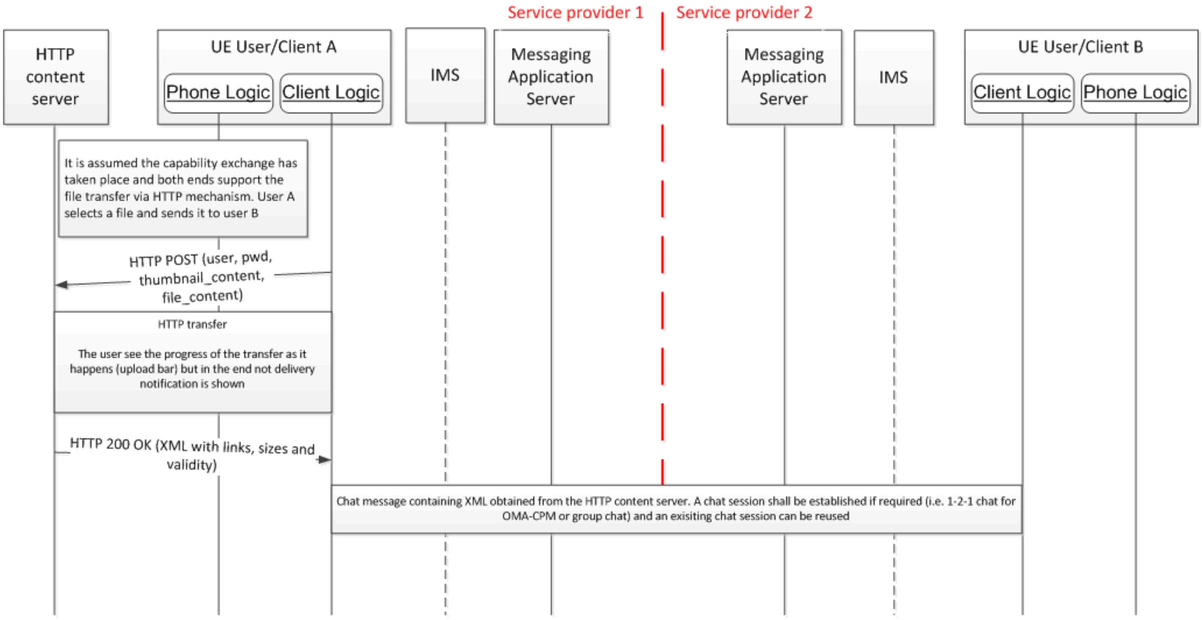

Figure 55: File transfer via HTTP: Sender procedures

Both the XML body returned by the HTTP Content Server and the optionally extended one that is exchanged between the clients shall correspond to following XML Schema which may be extended further by specific implementations and future versions of this specification. Such extensions shall be ignored by clients that are not aware of them:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:gsma:params:xml:ns:rcs:rcs:fthttp"
        xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns="urn:gsma:params:xml:ns:rcs:rcs:fthttp"
        elementFormDefault="qualified"
        attributeFormDefault="unqualified">
    <xs:element name="file">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="file-info" minOccurs="1" maxOccurs="2">
                    <xs:complexType>
                        <xs:sequence>
                            <xs:element name="file-size">
                                <xs:simpleType>
                                    <xs:restriction base="xs:integer"/>
                                </xs:simpleType>
                            </xs:element>
                            <xs:element name="file-name" minOccurs="0"
                            maxOccurs="1">
                                <xs:simpleType>
                                    <xs:restriction base="xs:string"/>
                                </xs:simpleType>
                            </xs:element>
                            <xs:element name="content-type">
                                <xs:simpleType>
                                    <xs:restriction base="xs:string"/>
                                </xs:simpleType>
                            </xs:element>
                            <xs:element name="data">
                                <xs:complexType>
```

```
                                                                                    <xs:attribute name="url"
                                                                                    type="xs:anyURI" use="required"/>
                                                                                    <xs:attribute name="until"
                                                                                    type="xs:dateTime" use="required"/>
                                                                                    <xs:anyAttribute
                                                                                    namespace="##other"
                                                                                    processContents="lax"/>
                                                                        </xs:complexType>
                                                            </xs:element>
                                                            <xs:any namespace="##other" processContents="lax"
                                                            minOccurs="0" maxOccurs="unbounded"/>
                                                </xs:sequence>
                                                <xs:attribute name="type" use="required">
                                                            <xs:simpleType>
                                                                        <xs:restriction base="xs:string">
                                                                                    <xs:enumeration value="file"/>
                                                                                    <xs:enumeration value="thumbnail"/>
                                                                        </xs:restriction>
                                                            </xs:simpleType>
                                                </xs:attribute>
                                                <xs:attribute name="file-disposition" use="optional">
                                                            <xs:simpleType>
                                                                        <xs:restriction base="xs:string">
                                                                                    <xs:enumeration value="render"/>
                                                                                    <xs:enumeration
                                                                                    value="attachment"/>
                                                                        </xs:restriction>
                                                            </xs:simpleType>
                                                </xs:attribute>
                                                <xs:anyAttribute namespace="##other" processContents="lax"/>
                                    </xs:complexType>
                        </xs:element>
                        <xs:any namespace="##other" processContents="lax" minOccurs="0"
                        maxOccurs="unbounded"/>
            </xs:sequence>
      </xs:complexType>
  </xs:element>
</xs:schema>
```

Table 39: File transfer via HTTP message body schema

3.5.4.8.3.1.1 Upload Resume

In case a file upload cannot be completed, e.g. because the file sender loses network coverage, the RCS client should allow to resume the File Transfer by using the procedure described in this section. It is intended to resume the upload of the file itself but not of an optional thumbnail which has small size. The content server shall store partial uploads and make them accessible via the related TID defined in 3.5.4.8.3.1. As it may apply a service provider policy and remove partially uploaded files after some time, resume upload may just be possible for a limited time. In case it fails, the upload cannot be resumed and the complete file needs to be uploaded again following the procedure in section 3.5.4.8.3.1. The following operations are used:

1. **Get upload info:** A client that intends to resume the upload of an interrupted File Transfer shall fetch the upload information of the file by a HTTP GET request to the content server including the TID related to the initial upload or former resume upload (see section 3.5.4.8.3.1).

GET <FT HTTP CS URI>?tid=<tid_value>&get_upload_info HTTP/1.1

The server sends back the upload information in the following XML structure describing the file content without optional thumbnail including the stored byte range within a file-range tag and the direct upload URI.

```
<?xml version="1.0" encoding="UTF-8"?>
<file-resume-info xmlns="urn:gsma:params:xml:ns:rcs:rcs:fthttpresume">
        <file-range start="[start-offset in bytes]" end="[end-offset in bytes]" / >
        <data url="[HTTP upload URL for the file]"/>
</file-resume-info>
```

Table 40: File transfer via HTTP upload information content

Complying with following schema:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:gsma:params:xml:ns:rcs:rcs:fthttpresume"
              xmlns:xs="http://www.w3.org/2001/XMLSchema"
              xmlns="urn:gsma:params:xml:ns:rcs:rcs:fthttpresume"
              elementFormDefault="qualified"
              attributeFormDefault="unqualified">
        <xs:element name="file-resume-info">
              <xs:complexType>
                    <xs:sequence>
                              <xs:element name="file-range">
                                        <xs:complexType>
                                                  <xs:attribute name="start" type="xs:integer"
                                                  use="required" />
                                                  <xs:attribute name="end" type="xs:integer"
                                                  use="required" />
                                                  <xs:anyAttribute namespace="##other"
                                                  processContents="lax"/>
                                        </xs:complexType>
                              </xs:element>
                              <xs:element name="data">
                                        <xs:complexType>
                                                  <xs:attribute name="url" type="xs:anyURI"
                                                  use="required"/>
                                                  <xs:anyAttribute namespace="##other"
                                                  processContents="lax"/>
                                        </xs:complexType>
                              </xs:element>
                              <xs:any namespace="##other" processContents="lax"
                              minOccurs="0"
                              maxOccurs="unbounded"/>
                    </xs:sequence>
              </xs:complexType>
        </xs:element>
</xs:schema>
```

Table 41: File transfer via HTTP upload information schema

In case of a successful HTTP response by the server, e.g. HTTP 200, including an XML description of the file, the following procedure applies depending on the content of the XML description:

- If it includes file-resume-info for the uploaded file content with file range which matches the original file size, the file has been uploaded successfully.

- If it includes file-resume-info of the uploaded file content but with file range below the file size, the remaining file content needs to be uploaded using step 2.
- If it does not include the file-resume-info of the file content, the full upload needs to be started from beginning using the HTTP POST request as described section 3.5.4.8.3.1.

NOTE: The file-range refers to the part of the file that has been uploaded prior to the resume upload.

A server shall send back an HTTP error response if resume upload cannot be performed (e.g. because the partial files are no longer available) according to [RFC2616], e.g. HTTP 404 or 410. An HTTP response that does not contain an XML description of the file or an XML structure that does not include a range field, shall indicate to the client that a resume of the upload of the file is not possible and that therefore a full upload needs to be done again.

2. **Resume upload**: In case the client wants to resume the upload of the file content it generates an HTTP PUT request to the upload URL that was included in the XML description provided by the content server in operation 1. In this request it shall provide the remaining bytes started from the already uploaded byte position that was included in the received XML description. To indicate the byte range that is included in the HTTP PUT request a HTTP *Content-Range* header as defined in [RFC2616] is added to the request:

```
PUT http://<file_upload_uri> HTTP/1.1

Content-Type: [mime type depending on the file; e.g. image/jpeg]

Content-Length: <remaining_upload_size>

Content-Range: bytes <first-byte-pos> - <last-byte-pos> / <file_size>
Authorization: Digest ...

<file content>
```

Table 42: File transfer via HTTP upload information content

When the server receives the partial file, it shall append the data according to the Content-Range header. In case the upload is successful, a HTTP 200 OK response without body is returned.
The client has to ensure that the file content related to the TID has not been changed between the initial HTTP POST request and the resume upload operation.

NOTE: This HTTP PUT can fail, e.g. due to another loss of network coverage. In that case, the operations 1 and 2 may be repeated with the same TID. In that case, the file-range tag indicates the sum of all the data uploaded in the uploaded resumes that have taken place so far.

3. **Get download info**: To get the XML description of the complete file to be sent to the file receiver according to 3.5.4.8.3.1, the client sends the following request to the content server:
GET <FT HTTP CS URI>?tid=<tid_value>&get_download_info HTTP/1.1
The server sends back a successful HTTP response including the XML description back if the file has been uploaded successfully. In that case the XML includes the file

info for the thumbnail (if provided) and the file (as defined in Table 37). Otherwise an HTTP error response will be returned.

NOTE: Like for the initial HTTP POST in section 3.5.4.8.3.1 authentication may be requested for other HTTP requests used in this section. In that case, the client shall sent it a second time carrying the authentication header field in line with the challenge received in the HTTP 401 AUTHENTICATION REQUIRED response to the first request. All additional HTTP requests towards the content server shall use the HTTP digest authentication as defined for regular file upload. In case HTTP over TLS is used, HTTP basic authentication can be used instead.

The whole procedure (including the initial upload is summarized in following figures:

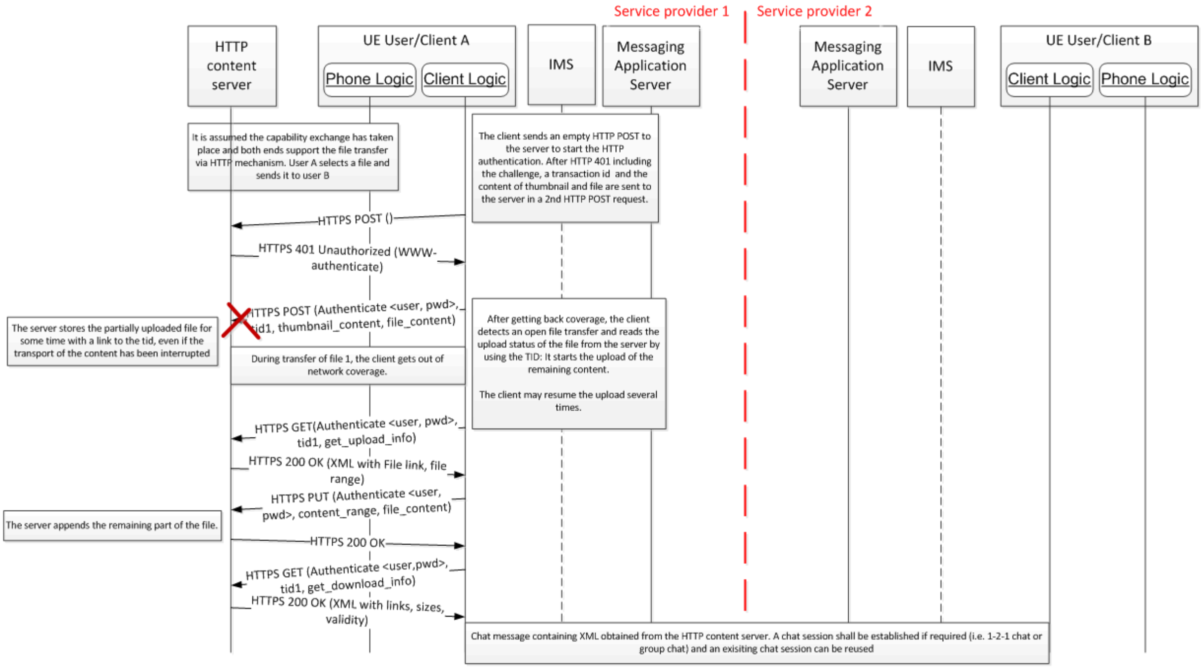

Figure 56: File transfer via HTTP: Resume upload

In case the resume is not possible (anymore), the flow shall be as follows:

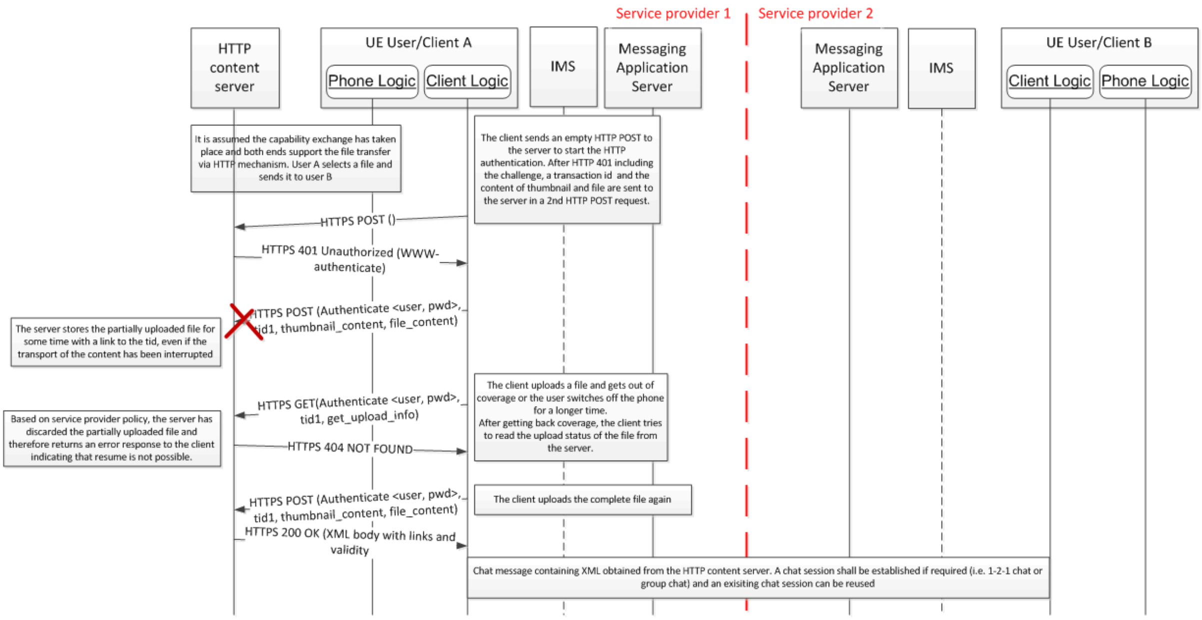

Figure 57: File transfer via HTTP: Resume upload not possible

3.5.4.8.3.2 Receiver procedures

When the receiver gets a chat message as described in the previous section, the RCS client shall:

1. The user shall not be aware a different procedure has been used to carry the file, therefore and, if present, the RCS client shall download (HTTPS GET) the thumbnail and display/notify of the incoming file transfer.
2. If the user accepts, the file shall be downloaded (HTTPS GET) showing the progress of the download as for a file transfer performed for MSRP. If the HTTP content server is working adequately, one of the following three responses shall be returned to the client:

    - HTTP 200 OK: Meaning the file is downloaded The client shall handle the file then according to the file-disposition attribute if included in the File transfer via HTTP message body content (see Table 38 and section 3.5.4.8.3.1).
    - A HTTP 503 INTERNAL SERVER ERROR with a Retry-After header: In this case the client shall retry, the recommended value to retry will be specified in the “Retry-After” header. Please note that this response is provided by the server when the sender is still uploading the file to prevent the race condition.
    - Any other error: The client shall retry up to a maximum of 3 times. In case the file was partially downloaded already, a partial HTTP GET request as defined in [RFC2616] may be used to obtain the remaining part of the file.

3. Regarding the display notification associated to this chat message, it shall only be sent when the file has been successfully downloaded to indicate the sender that the file has been effectively downloaded by the user.

Finally note that if validity of the file to be downloaded indicates that it may no longer be available on the server, the client shall inform the user of the circumstance when trying to download the file. The detailed UX is left intentionally outside the scope of this specification and it is up to the RCS client implementation.

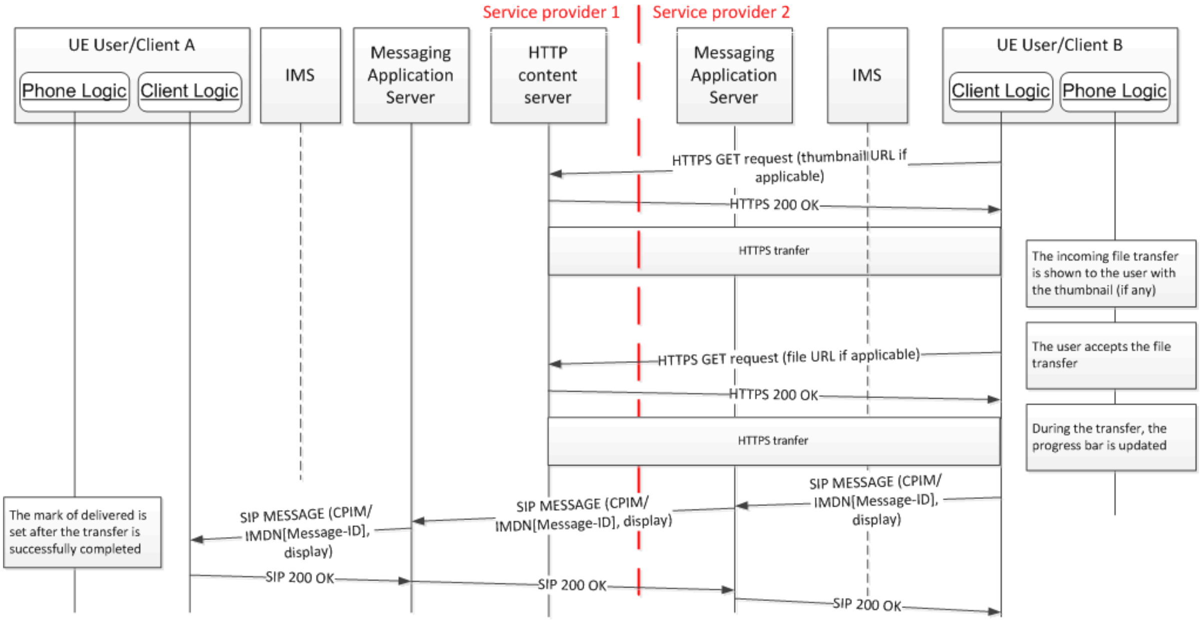

Figure 58: File transfer via HTTP: Receiver procedures

3.5.4.8.3.2.1 File transfer auto-accept

Consistently with Annex A sections A.1.5 and A.2.6, if the parameter *FT AUT ACCEPT* is set to 1 and the indicate file size is smaller than the size configured in the *FT WARN SIZE* configuration parameter, the receiving client shall not only download automatically the thumbnail but also the file content.

3.5.4.8.4 HTTP Content server addressing

In order to enable the traceability of the HTTP transactions among operators, the HTTP content server FQDN shall follow the format presented below:

ftcontentserver.rcs.mnc<MNC>.mcc<MNC>.pub.3gppnetwork.org

Table 43: HTTP content server FQDN

3.5.4.8.5 Security considerations

In order to guarantee the integrity and security of the solution for file transfer via HTTP the following three principles shall be taking into account:

1. The security of the solution relies on the security of the chat messages. Therefore, encryption of the media associated to Chat (1-to-1/Group Chat) media is recommended.
2. All HTTP transactions shall be secured using HTTPS.
3. To secure interoperability between Service Providers and to reduce complexity on the RCS device/client, the HTTP configuration server shall make use of public root certificates issued by a recognised CA. That is the root certificates are similar to those used by standard webservers which are widely recognised by browsers and web-runtime implementations both in PCs and devices.

3.5.4.8.6 File Transfer Fallback

When the configuration parameter FT HTTP FALLBACK defined in section A.1.5 is set to 1, an RCS client shall use the following procedure to send files and multimedia content to a contact that does not support any of the enabled File Transfer mechanisms (including a non RCS contact):

1. The RCS client shall upload the content to the HTTP Content Server as defined in steps 2 to 4 of section 3.5.4.8.3.1. A thumbnail shall not be provided in this case.
2. The RCS client shall extract the link and optionally the validity from the obtained XML body.
3. The RCS client shall compose a regular text message including the link and optionally the validity.

NOTE: The exact content of the message is out of scope of this specification.

4. The RCS client shall send this text message to the contact using its standard technology for sending text messages to such contacts (e.g. SMS).

3.5.4.8.7 HTTP State Management

The client should support for the HTTP procedures for File Transfer via HTTP state management defined in [RFC6265]. This includes all HTTP requests and responses between the client and content servers as part of the File Transfer procedure defined in section 3.5.4.8.3. This allows a content server to return in HTTP responses a Set-Cookie header. The client should apply the parsing and storage procedures of the Set-Cookie header as defined [RFC6265]. It should send the cookie header in HTTP requests to content servers respecting the cookie attributes provided by the server in the Set-Cookie header in accordance with [RFC6265].

With this the content server is able to make use of all the functions of HTTP state management.

NOTE: It is left to service provider policies to ensure that response does not include the cookie towards legacy clients.

3.5.4.9 Handling of specific content

3.5.4.9.1 Personal Card format

Current implementations of the vCard standard by different device manufacturers leads today to data loss of certain contact information, when this information is exchanged among devices or synced with network address books. An RCS compliant device shall support receiving at a minimum, vCard 2.1 [vCard21] and vCard 3.0 formats [RFC2425], [RFC2426] and may support also the Personal Contact Card (PCC) format [CAB_TS].

The following fields are considered key fields. No data of these fields should be lost when contact information is exchanged by any means (peer to peer contact sent, uploaded, synchronised, etc.):

- Name
- Telephone numbers
- Email addresses

- Address information
- Personal information

The Minimum subtypes that should be supported are defined in the PCC definition in [CAB_TS]:

- Name: Composed names (such as “Jean-Baptiste”) shall be supported properly.
- Personal Information:
  - Nickname
  - Photo
  - Birthdate
  - Comment
- Telephone number: At least the following subtypes of telephone number shall be supported:
  - Land home
  - Land work
  - Land other
  - Mobile home
  - Mobile work
  - Mobile other
  - Fax work
  - Fax other
  - Beeper
  - Other
- Email addresses: The following subtypes shall be supported:
  - Email work 1
  - Email work 2
  - Email home 1
  - Email home 2
  - Other
- Address information
  - Address
  - Geographic Position
  - Time zone

Sending and receiving a contact card via File Transfer is technically the same as sending any other file.

If the format for pushing a contact card file is vCard 2.1 or 3.0 formats, the MIME (Multipurpose Internet Mail Extensions) type that shall be used for the file transfer is “*text/vcard*”.

If the format for pushing the contact card is CAB (Converged Address Book) 1.0 PCC XML format, then the CAB PCC MIME type "*application/vnd.oma.cab-pcc+xml*" shall be used.

On the receiving side, after the receiving RCS user accepts the contact card file delivered through File Transfer, the receiving RCS client shall apply the mapping of the RCS supported fields between the received format (CAB PCC XML for example) and the used format of the local address book database[33].

vCard 3.0 format is recommended in RCS.

If the receiving side does not support the offered format identified in *the a=file-selector* attribute of the SIP INVITE SDP, it should reject the File Transfer invitation with an error response indicating it does not support the content-type, which then causes the sending side to initiate a second File Transfer, this time sending the contact card in a different format.

3.5.4.9.2 Audio Message

The handling of audio messages is described in section 3.11.4.

3.5.5 NNI and IOT considerations

In addition to what is defined in Section 2.12, the mapping of the appropriate File Transfer feature tags is done by the Messaging Server, as per Appendix G in [RCS-CPM-CONVFUNC-ENDORS] when it is determined that the remote network requires such interworking.

3.5.6 Implementation guidelines and examples

From the UX perspective there are five possible entry points to this service:

1. Address book/Call-log: A file transfer can be initiated with any registered contact providing the correct capabilities are in place. This is contact oriented initiation. Following the address book interaction, the list of available files is displayed allowing the user to select one or more files to share. Once the file transfer commences, the progress can be checked in the standard notification area.

[33] If the conversion between PCC and vCard is required, please see [CAB_TS] section 5.4.3 "Format Adaptation".

Figure 59: Reference UX for accessing file share from address book/call-log

2. Media gallery/File browser: The user can browse, select a file (or multiple files) and then share these with one or more RCS users. This is task contact oriented initiation. Only RCS capable users shall be displayed as candidate recipients of the file.

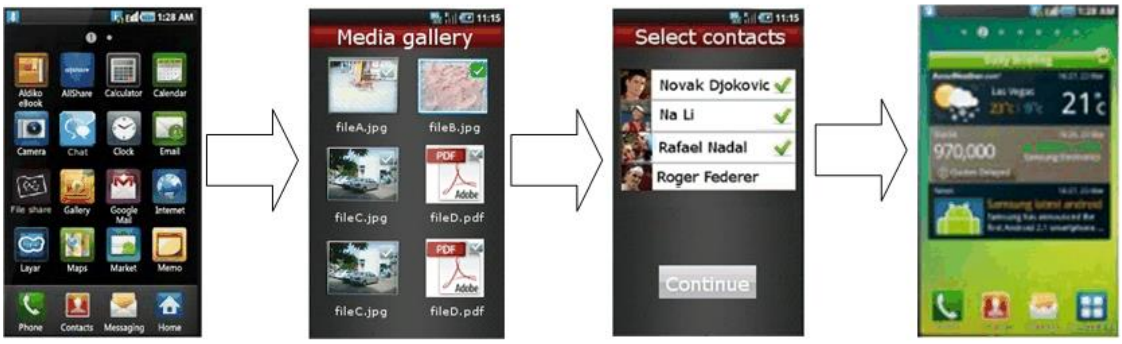

Figure 60: Reference UX for accessing file share from media gallery or file browser

In the previous figure, once File Transfer is selected, the user will be presented with the complete list of RCS contacts (including contacts which are currently not registered).
In this case, a SIP OPTIONS or Presence exchange is triggered once a contact is selected from the list.

3. Camera application: The experience is similar to the media gallery/file browser experience with the difference being that the user is able to select only the last picture or video (and, in some cases, a picture or video from the camera gallery) to be shared.
4. Chat window: From the Chat window a file can be shared using the relevant button/icon. The experience is identical to the address book/call-log. The user is redirected to the media gallery or file explorer where the user can choose a file which, is then shared with the conversation partner(s).

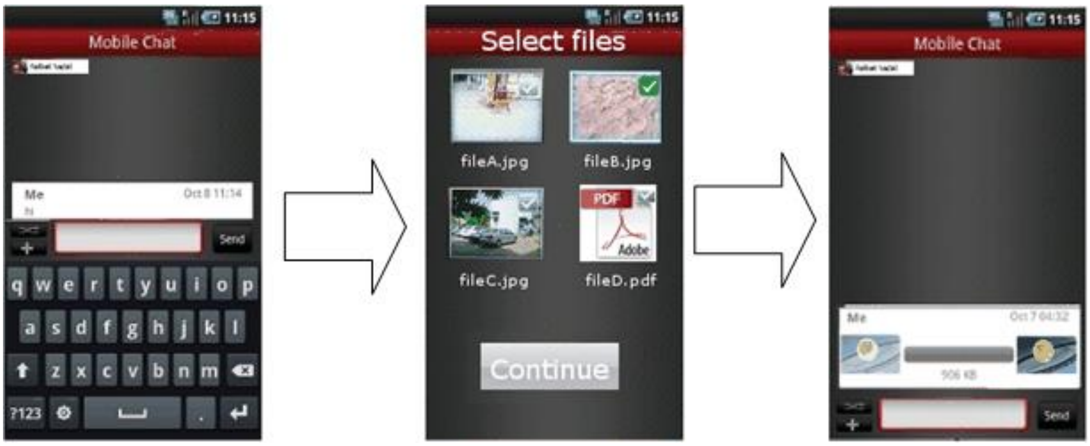

Figure 61: Reference UX for accessing file share from a Chat window

5. Call screen (Image Share): a picture can be shared either from the camera (front or back) or by choosing a file from the media gallery. Please note this case has been covered in detail in section 3.6.6.1.2.

3.5.6.1 Handling of specific content

3.5.6.1.1 Personal Card handling

The personal and business cards of the RCS user may be stored in a way that is compliant to the CAB 1.0 PCC data in the RCS client which enables the RCS user to create and populate any number views on the personal and/or business contact information as needed. A client may tag these with their dedicated purposes (professional, friends, etc.).

A Personal Card is, from a technical perspective, the same as any other contact card. This functionality only requires certain user experience changes. In particular:

- Visibility as an option in the address book menu.
- A special name/mark in the address book to easily distinguish it from the rest of the contacts.

It is recommended to support at least three Personal Cards. In particular:

- Business Card: For professional use.
- Two more Personal Cards to allow social uses (e.g., a contact card to be exchanged with closest friends for having fun, including frequently updated fields such as a personal picture) and an additional one to allow having a stable personal profile for non-professional uses.

3.5.6.1.2 Personal Contact Card entry points

Sending a contact card

The user selects any of the contacts in the address book. Entry points for sending a contact could be:

1. Chat
2. Address book
3. Call log

Before sending a contact card, the user should have the option to preview the information. The possibility of editing the information should be available so that filtering the contact information to be sent is also allowed. Once the contact information is confirmed, the contact card is sent over File Transfer.

Receiving a contact card

When a new contact card is received, the user is prompted to accept the file. Once accepted, two options are given:

1. Save contact card
2. View contact card

If ‘Save Contact information’ is chosen proper options will be given depending on whether the contact received already exists or not in the receiver’s address book. If it exists the existing contact information will be implemented with the additional information received.

3.5.6.1.3 Audio Messages

The entry points for audio messages are described in section 3.11.6.

3.6 Content sharing

3.6.1 Feature description

3.6.1.1 In-Call Image Share and Video Share

Content sharing provides the capability to share videos and pictures in near real-time. This functionality can be used in connection to a voice. When the receiving user has multiple devices the content sharing requests are sent to all those devices. Therefore, the user can decide to accept the shared content on a different device than the one they are using for the voice call if that device has better display capabilities for instance.

There can be different sources for the shared videos and pictures:

- The front camera (“me”)
- The rear camera (“what I see”)
- A file (“video streaming” or “sending of a stored image”)

A Service Provider configurable parameter allows the Service Provider to set the maximum duration of a Video Share session (see VS MAX DURATION in section A.1.6) and the max size of a file transferred during Image Share (see IS MAX SIZE in section A.1.6).

From the user experience perspective and assuming that the duration and size limitations are in place (i.e. the values are non-zero):

- When performing a video share, if the session duration (send or receive) is approaching the VS MAX DURATION, a warning notification could be displayed for the user.
- When performing a video share, if the session duration (send or receive) is longer than the VS MAX DURATION, a warning message will be displayed and the video share session will be cancelled (that is at protocol level, the SIP BYE request will be sent to the other end).

- If the picture size in an image share session is bigger than IS MAX SIZE, a warning message will be displayed when trying to send or receive a picture larger than the mentioned limit and the image share session will be cancelled (that is at protocol level, the SIP INVITE request will never be sent or an automatic rejection response will be sent to the other end depending on the scenario).

The in-call Video Share and Image Share services are linked to the call. Therefore, they are also automatically terminated when the call ends.

All services are delivered as one to one only and there is no multiparty sharing provided. For the content sharing during a call, the user should be able to recognise which content sharing services (e.g. Video and Share or Image Share) are available to use with their conversation partner. Therefore, both ends need to be updated on the respective capabilities to avoid showing a service as available when this is no longer the case. This is achieved through the capability exchange described in section 2.6.

Video Share and Image Share are unidirectional services and do not need a dedicated audio path. It is possible to establish simultaneous Image and/or Video Share sessions in each direction. For example when referring to bidirectional Video Share, this means that once User A is sharing video with User B, User B can also start to share video with User A simultaneously, provided the right coverage conditions are in place. In this case, each Video Share session is independent and should be handled separately. When a user’s coverage conditions change while such bi-directional sharing is active, the device that changed coverage shall terminate the sharing session that it initiated. The same example would also apply to Image Share or to a combination of Video Share in one direction and Image Share in the other.

3.6.1.2 In-Call Shared Map and Shared Sketch

This provides the capability to share maps and drawing canvas in near real-time. This functionality can be used during a voice call.

3.6.1.2.1 Shared Map

The Shared Map application lets two users draw, share markers and view each other’s position on a “shared” map. The service description is provided in section 1.1 of [PRD-RCC.20].

3.6.1.2.2 Shared Sketch

The Shared Sketch application lets two users, draw, add background images, change background colour on a “shared” canvas. The service description is provided in section 1.1 of [PRD-RCC.20].

3.6.1.3 Call Composer and Post-call service

This provides the capability to share multi-media content ahead of the call to provide context to the called party when the call is set up and to share a note (reason) or a voice message after an unanswered call.

3.6.1.3.1 Call composer

For sharing pre-call rich content related to a voice call, the descriptions in section 1.1 of [PRD-RCC.20] shall be taken into consideration.

3.6.1.3.2 Post-Call service

For sharing a note (reason) or a voice message after an unanswered call, the descriptions in section 1.1 of [PRD-RCC.20] shall be taken into consideration.

3.6.1.4 Use Cases

3.6.1.4.1 Share Video during a voice call

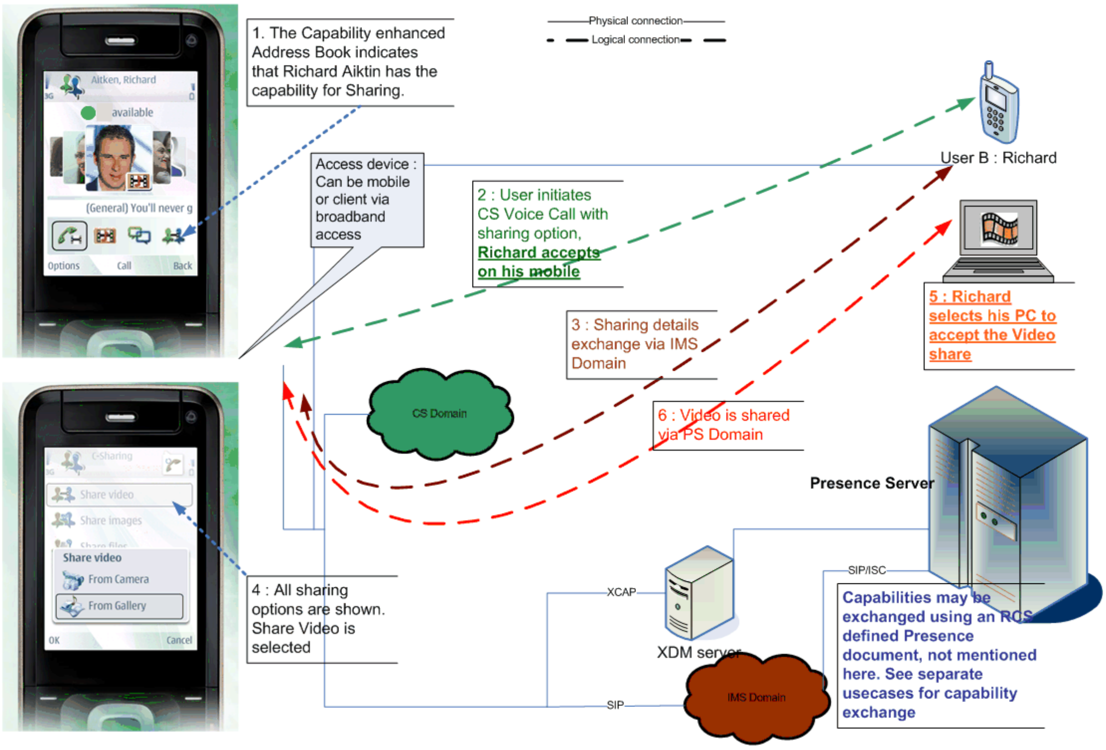

Figure 62: Sharing video during a voice call

Figure 62 illustrates the behaviour when the voice call is set up in the CS domain. Apart from the voice call itself, the behaviour would be identical though if one or both parties used the PS domain for the voice call as specified in section 3.8 since the sharing service functions independently of the voice call.

NOTE: When both of the devices involved in the sharing are on a high bandwidth access, for example LTE, the perceived video quality will be higher.

3.6.1.4.2 Sharing video during a call in the multidevice environment

User A has a mobile device and a broadband access device (RCS PC client). User B has a mobile device.

- User B has travelled to Hong Kong and is visiting the Victoria's peak. The view from top of the peak is astonishing and they would like to share the experience with their friend User A.
- User B makes a call to User A.

- User A answers on the mobile.
- User B tells User A about the view they are viewing. To prove this User B decides to share a video with User A.
- User B sees from the call menu that they can share video with User A. User B sends the request to share video, for example, by clicking the Video Share icon.
- The request is sent to both User A's mobile and PC; both mobile and PC will alert.
- As User A is sitting in front of their PC he/she decides to take the video to the PC for example, by clicking accept button on the PC client.
- User A's mobile will then stop alerting.
- User A will now see the beautiful scenery shared by User B in their PC while still having the voice call on the mobile.

NOTE: This was illustrated previously in Figure 62. The behaviour would be similar when sharing an image.

3.6.1.4.3 Share an Image during a call

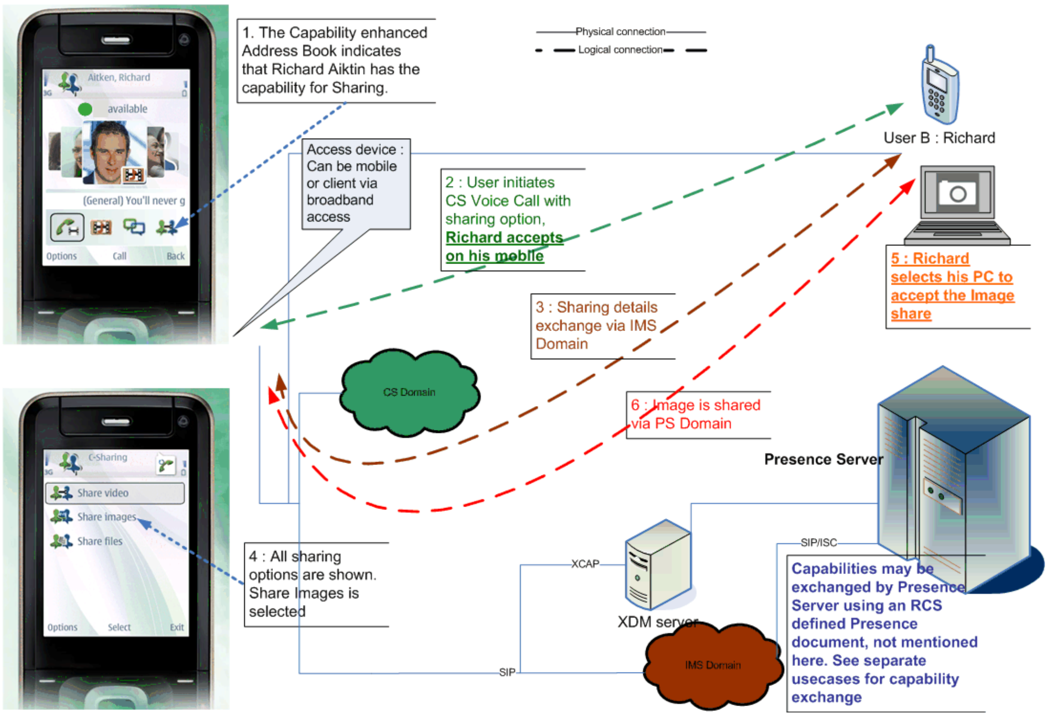

Figure 63: Sharing an image during a call

Figure 63 illustrates the behaviour when the voice call is set up in the CS domain. Apart from the voice call itself, the behaviour would be identical though if one or both parties used the PS domain for the voice call as specified in section 3.8 since the sharing service functions independently of the voice call.

3.6.2 Interaction with other RCS features

3.6.2.1 Voice Call

The Video Share, Image Share, Shared Map and Shared Sketch services during a voice call (either over CS or as specified in section 3.8) interacts with that voice call since the sharing is automatically terminated when the call is terminated. There is also an interaction with the supplementary services of that voice call.

NOTE: This interaction does not apply for the File Transfer and 1-to-1 chat service. The sharing session is independent of that voice call and progresses independently of the voice call continuity.

3.6.2.1.1 Multiparty call and In-Call sharing services

Once a voice call is established between two users, it is possible for one of them to add another party to the call, and consequently, initiate a multiparty call. From RCS services perspective and as presented in section 2.7, the Image Share, Video Share, Shared Map and Shared Sketch services are not available during a multiparty call. Therefore, the terminal should manage the following scenarios:

- The users were in a voice call without using the Image Share, Video Share, Shared Map or Shared Sketch services: In this case, when switching to a multiparty call the client starting the process has to send a SIP OPTIONS request with a capability update (as described in section 3.6.4.7.2) indicating that the Content Sharing services during a call are no longer available. The on-screen icons/layout should be updated accordingly.
- The users (User A and User B) were in a voice call using Video Share: In this case, switching to a multiparty call means ending the Video Share service. This can either be sender or receiver terminated, depending upon the circumstances, as described in sections 3.6.4.7.4 and 3.6.4.7.5 respectively. In both cases, a capabilities exchange using SIP occurs and, consequently, the client initiating the multiparty call should report that the Content Sharing services/capabilities during a call are no longer available. The on-screen icons/layout should be updated accordingly.
- The users (User A and User B) were in a voice call using Image Share with the transfer not yet completed: In this case, switching to a multiparty call means ending the Image Share service. This either can be sender or receiver terminated, depending upon the circumstances, as described in sections 3.6.4.7.8 and 3.6.4.7.9 respectively. In both cases, a capabilities exchange using SIP OPTIONS occurs and, consequently, the client initiating the multiparty call should report that the Content Sharing services/capabilities during a call are no longer available. The on-screen icons/layout should be updated accordingly.
- The users (User A and User B) were in a voice call using Image Share after the transfer has completed: In this case, switching to a multiparty call means that the picture is no longer shown in the call screen and that the client starting the process has to send a SIP OPTIONS message with a capability update (as described in section 3.6.4.7.2) indicating that the Content Sharing services during a call are no longer available. The on-screen icons/layout should be updated accordingly.
- The users (User A and User B) were in an active Shared Map session: In this case, switching to a multiparty call means ending the Shared Map session. This can be initiated by either user (user A or user B) depending upon the circumstances. A capabilities exchange using SIP occurs and, consequently, the client initiating the multiparty call should report that the Content Sharing services/capabilities during a call are no longer available.
- The users (User A and User B) were in an active Shared Sketch session: In this case, switching to a multiparty call means ending the Shared Sketch session. This can be initiated by either user (user A or user B) depending upon the circumstances. A capabilities exchange using SIP occurs and, consequently, the client initiating the

multiparty call should report that the Content Sharing services/capabilities during a call are no longer available.

It should be also noted that from the moment the users enter in a multiparty call, it is not necessary to perform the capability exchange described in section 3.6.4.7.2.

Finally, if the multiparty call is converted into a standard call (That is it becomes again a 1-to-1 call), this event should be treated as a new call establishment meaning that a capability exchange via OPTIONS needs to take place and, consequently, the relevant on screen icons need to be updated.

3.6.2.1.2 Call on hold and In-Call sharing services

Once a voice call is established between two users, it is possible for one of them to put the other party on hold. From RCS services perspective and as presented in section 2.6.3.1, the Image Share, Video Share, Shared Map and Shared Sketch services are not available during a call which is not active, therefore, the terminal needs to manage the following scenarios:

- The users were on a voice call without using the Image Share, Video Share, Shared Map or Shared Sketch services: In this case, when putting the call on hold the client starting the process has to send an SIP OPTIONS request with a capability update (as described in section 3.6.4.7.2) indicating that the Content Sharing services during a call are no longer available. The on-screen icons/layout should be updated accordingly.
- The users (User A and User B) were in a voice call using Video Share: In this case, putting the call on hold means ending the Video Share service. This can either be sender or receiver terminated, depending upon the circumstances, as described in sections 3.6.4.7.4 and 3.6.4.7.5 respectively. In both cases, a capabilities exchange using SIP OPTIONS occurs and, consequently, the client putting the call on hold should report that the Content Sharing services/capabilities during a call are no longer available. The on-screen icons/layout should be updated accordingly.
- The users (User A and User B) were in a voice call using Image Share with the transfer not having completed: In this case, putting the call on hold means ending the Image Share service. This can either be sender or receiver terminated, depending upon the circumstances, as described in sections 3.6.4.7.8 and 3.6.4.7.9 respectively. In both cases, a capabilities exchange using SIP OPTIONS occurs and, consequently, the client putting the call on hold should report that the Content Sharing services/capabilities during a call are no longer available. The on-screen icons/layout should be updated accordingly.
- The users (User A and User B) were on a voice call using Image Share after the transfer has completed: In this case, putting the call on hold means that the picture is no longer shown in the call screen and that the client starting the process has to send a SIP OPTIONS message with a capability update (as described in section 3.6.4.7.2) indicating that the Content Sharing services during a call are no longer available. The on-screen icons/layout should be updated accordingly.
- The users (User A and User B) were on an active Shared Map session: In this case, putting the call on hold means ending the Shared Map session. This can be initiated by either user (User A or User B) depending upon the circumstances. In both cases,

a capabilities exchange using SIP OPTIONS occurs and, consequently, the client putting the call on hold should report that the Content Sharing services/capabilities during a call are no longer available.
- The users (User A and User B) were on an active Shared Sketch session: In this case, putting the call on hold means ending the Shared Sketch session. This can be initiated by either user (user A or user B) depending upon the circumstances. In both cases, a capabilities exchange using SIP OPTIONS occurs and, consequently, the client putting the call on hold should report that the Content Sharing services/capabilities during a call are no longer available.

It should also be noted that from the moment the call is put on hold (that is the call is not active):

- It is not necessary to perform the capability exchange described in section 3.6.4.7.2, and,
- If there is another active call, the behaviour regarding the Image Share, Video Share, Shared Map and Shared Sketch services (that is both for the capability exchange and the services itself) should not be affected by the fact that another call is on hold.

Finally, if the call is made active, this event should be treated as a new call establishment meaning that a capability exchange via OPTIONS needs to occur and, consequently, the relevant on screen icons need to be updated.

3.6.2.1.3 Waiting call and In-Call sharing services

A waiting call is a non-active call; therefore, consequently with the information presented in section 2.6.3.1, it should not be possible to access the Image Share, Video Share, Shared Map and Shared Sketch services between the caller and receiver.

Please note having a waiting call will not affect the behaviour for Image Share, Video Share, Shared Map and Shared Sketch services (that is both for the capability exchange and the services itself) on the active call.

3.6.2.1.4 Calls from private numbers

When a call is received and the caller cannot be identified (because a hidden number is used for instance), it should not be possible to access the Image Share, Video Share, Shared Map and Shared Sketch services between the caller and receiver.

3.6.2.1.5 Call divert/forwarding

A receiver may have call divert/forwarding active (the calls are for instance forwarded to another number or to voicemail), it is still possible to access the Image Share, Video Share, Shared Map or Shared Sketch services from the caller to the receiver if, as per section 7.3.1.2 of [3GPP TS 24.279]:

- The caller has received a P-Asserted-Identity value from the receiver, or
- The caller has received a Connected Number information element and implements the procedure from section 7.3.1.2 of [3GPP TS 24.279].

Otherwise, it is not possible to access the Image Share, Video Share, Shared Map and Shared Sketch services from the caller to the receiver.

3.6.2.2 Video call

Please refer to section 3.9.2.2.

3.6.2.3 File Transfer

For sharing files the File Transfer service as described in section 3.5 is used.

3.6.3 High Level Requirements

3.6.3.1 Image Share and Video Share requirements

3-6-1 Image and Video content can be shared while on a CS or PS Voice call, thus it must be possible to have a voice and a data stream running simultaneously.

3-6-2 Each time a voice call is established, the user shall be offered the possibility to share image or video content whenever possible.

3-6-3 Void

3-6-4 Void

3-6-5 The Image Share and Video Share services shall be unidirectional. During a single Image or Video Share session, the originator of the session can share image or video content with the terminating party, but the terminating party cannot share image or video content with the originator in the same session.

3-6-6 The receiving party may be offered the possibility to establish a session in the other direction when circumstances allow.

3-6-7 Void

3-6-8 The Image Share and Video Shareservice can be initiated by either end point involved in the voice call (e.g. the caller or the receiver). When a user initiates image or video sharing, an invitation is automatically sent to the other contact, which may be accepted or rejected. An acceptance shall stand for all the contents shared during the call.

3-6-9 For the Image Share and Video Share services, End of communication shall be handled as follows:

- Image or video sharing session termination shall not lead to voice termination.
- Voice call termination shall automatically terminate the sharing session.

3-6-10 The receiver shall have the possibility to save the shared image or video on his/her device if allowed by the sender.

3-6-11 It shall be possible to assign a Service Provider configurable maximum content size allowed to be sent in an Image Share session. This enables the Service Provider of the inviting user’s RCS client to control the maximum size of the content that the inviting user’s RCS client is authorised to send in an Image Share session. The limitation should be transparent to the end-user.

3-6-12 It shall be possible to assign a Service Provider configurable maximum duration time allowed for a Video Share session. This enables the Service Provider of the inviting user’s RCS client to control the maximum duration time of a Video Share session that the inviting user’s RCS client is authorised to handle the limitation should be transparent to the end-user.

3-6-13 Void

3-6-14 Void

3-6-15 Void

3-6-16 Void

3-6-17 Void

3-6-18 It shall be possible for a terminating party or an originating party to terminate the Image Share or Video Share session.

3.6.3.2 Shared Map and Shared Sketch requirements

3-6-19 Maps and drawing canvas can be shared while on a CS or PS Voice call, thus it must be possible to have a voice and a data stream running simultaneously.

3-6-20 Each time a voice call is established, the user shall be offered the possibility to share map or drawing canvas content whenever possible.

3-6-21 Shared Map and Shared Sketch services shall be bidirectional. During a single Shared Map or Shared Sketch session, the originator of the session can share content with the terminating party and the terminating party can share content with the originator using the same session.

3-6-22 The Shared Map and Shared Sketch service can be initiated by either end point involved in the voice call (e.g. the caller or the receiver). When a user initiates map or drawing canvas sharing, an invitation is automatically sent to the other contact, which may be accepted or rejected. An acceptance shall stand for these contents shared during the call.

3-6-23 For Shared Map and Shared Sketch service, end of communication (case of active session while on a voice call) shall be handled as follows:

- Shared Map or Shared Sketch session termination shall not lead to voice termination.
- Voice call termination shall automatically terminate the Shared Map or Shared Sketch session.

3-6-24 The sender and the receiver should not save the last status of the map shared on his/her device.

3-6-25 The sender and the receiver shall have the possibility to save the last status of the sketch shared on his/her device.

3.6.3.3 Call composer and post-call service requirements

3-6-26 It shall be possible to the caller to “compose” pre-call information prior to placing the call and sharing it with the call receiver.

3-6-27 It shall be possible to the call receiver to see the composed pre-call information while receiving the incoming call.

3-6-28 The pre-call information can be picture file, subject, importance and location.

3-6-29 It shall be possible to the caller to “compose” additional information when a call is rejected or unanswered, for the other party to view.

3-6-30 It shall be possible to the party that missed or rejected the call to see the composed post-call information.

3-6-31 The post-call information can be a note (reason) or a voice message.

3.6.4 Technical Realisation

3.6.4.1 Video Share

Video Share during a voice call shall follow [PRD-IR.74] and take into account the handling of service capabilities and OPTIONS queries defined in sections 2.6.4.1 and 2.6 respectively. Furthermore, to allow the user to accept the sharing on any device a Broadband Access client (see section 2.9.1.4) shall not automatically reject the INVITE request if it is not in a voice call with the sender. It shall, therefore, alert the user as if it was and handle the user’s response as specified in section 2.11.

Interworking with Video Share terminals based on legacy specifications (i.e. the “already deployed terminals” option in [PRD-IR.74]) is not applicable in RCS.

[PRD-IR.74] mandates that the UE shall populate the P-Preferred-Service header and the network shall populate the P-Asserted-Service header with the Video Share ICSI "*urn:urn-7:3gpp-service.ims.icsi.gsma.videoshare*". The S-CSCF or AS that performs the service assertion in the originating network shall add the P-Asserted-Service header field set to the value of the asserted Video Share ICSI to SIP requests carrying the *+g.3gpp.cs-voice* media feature tag in the Accept-Contact header and remove the P-Preferred-Service header field (if present) before further routing the request.

NOTE: During a transition period towards full compliance, the network support for asserting the service is recommended but not mandatory.

A receiving network element and RCS client should ignore any SIP header fields that they do not understand (e.g. P-Preferred-Service, or P-Asserted-Service header fields).

3.6.4.1.1 Content Share Recording

A new SDP attribute in the media level is defined to be used by the Video or Image Share sender to indicate, in the SIP INVITE, to the recipient RCS client that the shared media can be recorded/saved.

The new SDP attribute as defined in [IETF-DRAFT-SIPREC-PROTOCOL]:

a=recordpref-attr = "a=recordpref:" pref where pref is set to "nopreference"

An SDP example is *a=recordpref:nopreference*

If the shared media in a Video or Image Share session is allowed (determined by the sender) to be recorded/saved, the sender RCS client should include the above SDP attribute in the SIP INVITE toward the recipient when setting up the Video or Image Share session. If the shared media in a Video or Image Share session shall not be saved by the recipient RCS client the sender RCS client shall not include the above SDP attribute in the SIP INVITE.

A Service Provider can provision its RCS clients to not always include this SDP attribute in the SIP INVITE setting up the Video or Image Share session so the shared media will not be recorded/saved by the recipient RCS client.

If the new SDP attribute is included in the SIP INVITE setting up the Video or Image Share session, it is to the decision of the recipient RCS client (under the instruction of the recipient user or user preference) to determine if the shared media will be recorded/saved.

3.6.4.1.2 Video Interoperability and encoding requirements

As presented in section 2.6.4.1, the Video Share service availability is mainly dependent on the network coverage. This is based on the assumption that both ends (source and destination) share the ability of handling a common video format and specific profile.

To guarantee the interoperability of RCS clients during Video Share scenarios, all RCS devices supporting the Video Share service shall, at least, support the following video format:

- Video format: H.264/MPEG-4 (Moving Pictures Experts Group) Part 10 // AVC (Advanced Video Codec).

- H.264 Profile: Baseline Profile (BP).
- H.264 Level: 1b[34]

NOTE: Please note that including this, it is highly recommended to support also the H.263-2000 codec with profile 0 Level 45 which is mandatory in RCS Release 1-4 Video Share that is based on [PRD-IR.74].

Next to these mandatory codecs, it is recommended to support additional video formats providing different levels of quality and to use them in an adaptive fashion depending both on the terminal status and the network conditions/coverage. As specified [RFC3264], formats must be listed in order of preference in the SDP media description. As such, additional codecs providing better quality than these mandatory ones should be listed in the SDP before the mandatory codecs. In any case for the encoding of the actual stream should be adapted to the currently available bandwidth and might, therefore, use bitrates lower than the maximum negotiated during session setup. To support this RTCP Receiver Reports (RR) shall be sent at least at a rate of one RR per second.

NOTE: In H.264, support of a certain level implicitly requires support for all lower levels, so a client supporting other H.264 levels should only indicate the highest level per profile that it supports.

Should an RCS terminal support several profiles, the final choice should be based on the outcome of the SDP media negotiation where both ends (sender and receiver) will present the supported video formats at that particular point (that is taking into account each device and network/connectivity status).

RTP payload handling shall be as described in section 7.4.3 of [3GPP TS 26.114] for the H.264 (AVC) video codec.

The receiving clients should preserve the aspect ratio of the incoming video stream, avoiding that video is stretched to fit the UI. The sending client may redefine the aspect ratio when supporting a flexible handling interface that could alternate between landscape and portrait (e.g. from 4:3 to 3:4 after the sending device has been rotated).

The originator of the Video Share session can indicate support for both Baseline (BP) and Constrained Baseline (CBP) Profiles.

The originator shall never use Flexible Macroblock Ordering (FMO), Arbitrary Slice Ordering (ASO), or Redundant Slices (RS) features of the profile no matter what the receiving client selects.

When a receiving client that supports both CBP and BP receives the combination of BP and CBP profiles within the same SDP offer it shall select CBP profile when the CBP media format is listed first in the SDP m-line.

When the SDP negotiation results in the use of the Baseline Profile (BP), a client shall not send Single-Time Aggregation Packet type A (STAP-A) packets, even when the

[34] The H.264 baseline profile 1b shall be encoded using the profile-level-id set to 0x42900B and the H.264 Constrained Baseline Profile 1b is 0x42D00B

packetization-mode 1 has been negotiated. When accepting the use of the Constrained Baseline Profile (CBP) a client shall support the use of STAP-A packets when packetization-mode 1 was negotiated.

```
v=0
o=- 1323909835 1323909838 IN IP4 x.x.x.x
s=-
c=IN IP4 x.x.x.x
t=0 0
m=video 4284 RTP/AVP 118 119
a=sendrecv
a=rtpmap:118 H264/90000
a=fmtp:118 packetization-mode=1;profile-level-id=42d00b
a=rtpmap:119 H264/90000
a=fmtp:119 packetization-mode=1;profile-level-id=42900b
```

Table 44: Example of a VideoShare SDP Offer with both CBP and BP profiles, each using Level 1b, preference for CBP since it is first in m-line

Coordination of Video Orientation (CVO) as specified in 3GPP Release 12 [3GPP TS 26.114] shall be supported with two (2) bits granularity by the UE and the entities in the IMS core network which terminate the user plane. The support for CVO shall be included in SDP offer and SDP answer as specified in section 6.2.3 of [3GPP TS 26.114].

3.6.4.1.3 Video Interoperability in LTE/HSPA

Video Share used over high bandwidth connections such as LTE or HSPA allows high bitrate bearers, thus allowing better user experience e.g. when using a large screen.

As specified in [PRD-IR.74], an RCS device shall support the H.264 video codec with baseline (and optionally Constrained Baseline Profile) profile and level 1.3[35] to provide 768 kilobits per second (kbps) video over an LTE bearer or over a similar high bitrate bearer. Please note that this codec is considered in addition to the mandatory formats specified in section 3.6.4.1.2.

If a second Video Share session is established in parallel, the H.264 video codec with baseline profile (and optionally Constrained Baseline Profile) and level 1.2[36] shall be used instead. The assumption for the use of a high bitrate bearer is that the connectivity and video parts of both terminals support it and have LTE or another high bitrate broadband access; otherwise the video bitrate will be reduced to the level 1b (as presented in section 3.6.4.1.2) to assure compatibility.

Also, in this case the encoding of the actual stream should be adapted to the currently available bandwidth and might therefore use bitrates lower than the maximum negotiated during session setup. Furthermore, in this case CVO as specified in 3GPP Release 12 [3GPP TS 26.114] shall be supported with two (2) bits granularity by the UE and the entities

---

[35] The H.264 baseline profile 13 shall be encoded using the profile-level-id set to 0x42800D. For H.264 CBP level 1.3 it is 0x42C00D.

[36] The H.264 baseline profile 12 shall be encoded using the profile-level-id set to 0x42800C. For H.264 CBP level 1.2 it is 0x42C00C.

in the IMS core network which terminate the user plane. The support for CVO shall be included in SDP offer and SDP answer as specified in section 6.2.3 of [3GPP TS 26.114].

3.6.4.1.4 Video Share duration

A configurable parameter allows the Service Provider to set the maximum duration of a Video Share session (see VS MAX DURATION in section A.1.6) in the UE. When one of the UEs which are sharing a live video stream detects that the maximum duration is reached, it shall tear down the Video Share session by sending a SIP BYE request. When sharing a live video stream, if the sharing duration (send or receive) is approaching the duration limitation, a warning notification could be displayed for prompting the two UEs. When sharing a stored video, if the UE detects that the video file being shared exceeds the Service Provider's configured maximum duration, it shall either not set up the session or tear it down depending on whether it is the initiator or the receiver.

3.6.4.2 Image Share

Image Share during a voice call shall follow [PRD-IR.79] where the SIP OPTIONS query shall be handled as specified in section 2.6 of this document. Furthermore, to allow the user to accept the sharing on any device a broadband access client (see section 2.9.1.4) shall not automatically reject the SIP INVITE request if it is not in a voice call with the sender. It shall alert the user and handle the user's response as specified in section 2.11.

To ensure that the request is sent to all devices with equal priority, clients using a PS voice service as defined in section 3.8 shall include the *+g.3gpp.cs-voice* feature tag in the Accept-Contact and Contact headers of the SIP INVITE request for content sharing. As described in [PRD-IR.79] the Image Share IARI is also included in the Accept-Contact and Contact headers (see Table 6 in section 2.6.1.1.2).

If the UE detects that the file being transferred exceeds the Service Provider configured maximum size (see IS MAX SIZE in section A.1.6), it shall either not set up the session or tear it down depending on whether it is the initiator or the receiver.

NOTE: All RCS services using MSRP, including Image Share, shall align with MSRP usage as described in section 2.8.

Details for image format as specified in [3GPP TS 26.141] will be followed.

[PRD-IR.79] mandates that the UE Shall populate the P-Preferred-Service header and the network shall populate the P-Asserted-Service header with the OMA CPM Standalone Messaging – Large Message Mode ICSI "*urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg*". The S-CSCF or AS that performs the service assertion in the originating network shall add the P-Asserted-Service header field set to the value of the asserted CPM Standalone Messaging – Large Message Mode ICSI to SIP requests carrying the *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-service.ims.iari.gsma-is"* IARI feature tag and value in the Accept-Contact header and remove the P-Preferred-Service header field (if present) before further routing the request.

NOTE: During a transition period towards full compliance, the network support for asserting the service is recommended but not mandatory.

A receiving network element and RCS client should ignore any SIP header fields that they do not understand (e.g. P-Preferred-Service, or P-Asserted-Service header fields).

3.6.4.3 Call composer

The technical realisation is based on procedures covered in sections 2.3 and 2.4 of [PRD-RCC.20].

3.6.4.4 Post-call service

The technical realisation is based on procedures covered in sections 2.3 and 2.5 of [PRD-RCC.20].

3.6.4.5 Shared Map

The technical realisation is based on procedures covered in section 2.9.7 and 2.9.9 and 2.9.10 of [PRD-RCC.20].

3.6.4.6 Shared Sketch

The technical realisation is based on procedures covered in section 2.9.8, 2.9.9 and 2.9.10 of [PRD-RCC.20].

3.6.4.7 Image Share and Video Share flows

3.6.4.7.1 General assumptions

In the following sections, we will show the relevant message flows and reference UX. Please note that the following assumptions have been made:

- For simplicity, the internal mobile network interactions are omitted in the diagrams shown in the following sections.
- The terminal and the network support 2G DTM and it is, therefore, always possible to gracefully terminate the content sharing session related to a voice call provided the terminal remains switched on. If 2G DTM is not supported, the case where on one of the ends a handover occurs to 2G would be result in behaviour towards the other end and the network that is equivalent to the one described for the case of a client error.
- The device is in coverage supporting bi-directional Video Share (see section 2.7). If this were not the case, additional capability exchanges would be required when starting and terminating a sharing session to indicate respectively that the device cannot handle an incoming Video Share session and that it can handle such an incoming Video Share session again.
- The terminal comes with a front and rear camera. If one or both are missing, the user should be notified only with the available options.
- Prior to a voice call, the user accessed the client's address book, call log or dial-pad to make the call. As described in section 2.6, while these actions are performed a capability query is executed to double-check on the available capabilities. As in older RCS versions including in some non RCS clients, Video and Image Share services are only available in a call, an OPTIONS exchange is required once the call is established to check on the capabilities during a call. This exchange can be initiated by either the sender or the receiver. In the following diagrams, it is assumed that this initial exchange (OPTIONS and response) has already taken place, and therefore, both ends are aware of the capabilities and the available RCS services.

- In the diagrams, it is assumed for simplicity that MSRP chunking is enabled. This is for representation purposes and it is up to the OEM to decide whether MSRP chunking is enabled or not.
- The flows in Figure 66, Figure 67, Figure 68, Figure 69, Figure 70, Figure 71, Figure 72, Figure 74, Figure 75 and Figure 76 show an OPTIONS exchange at the end of the flow. If the capability exchange is done using Presence, the equivalent Presence mechanism will be used.

3.6.4.7.2 Exchange capabilities during a call

The assumptions in this case are that User A and B are on a call when the capabilities of one of the users change (due to a hand-over to a different data carrier for instance). Therefore, the other end has to be informed using the OPTIONS message[37]

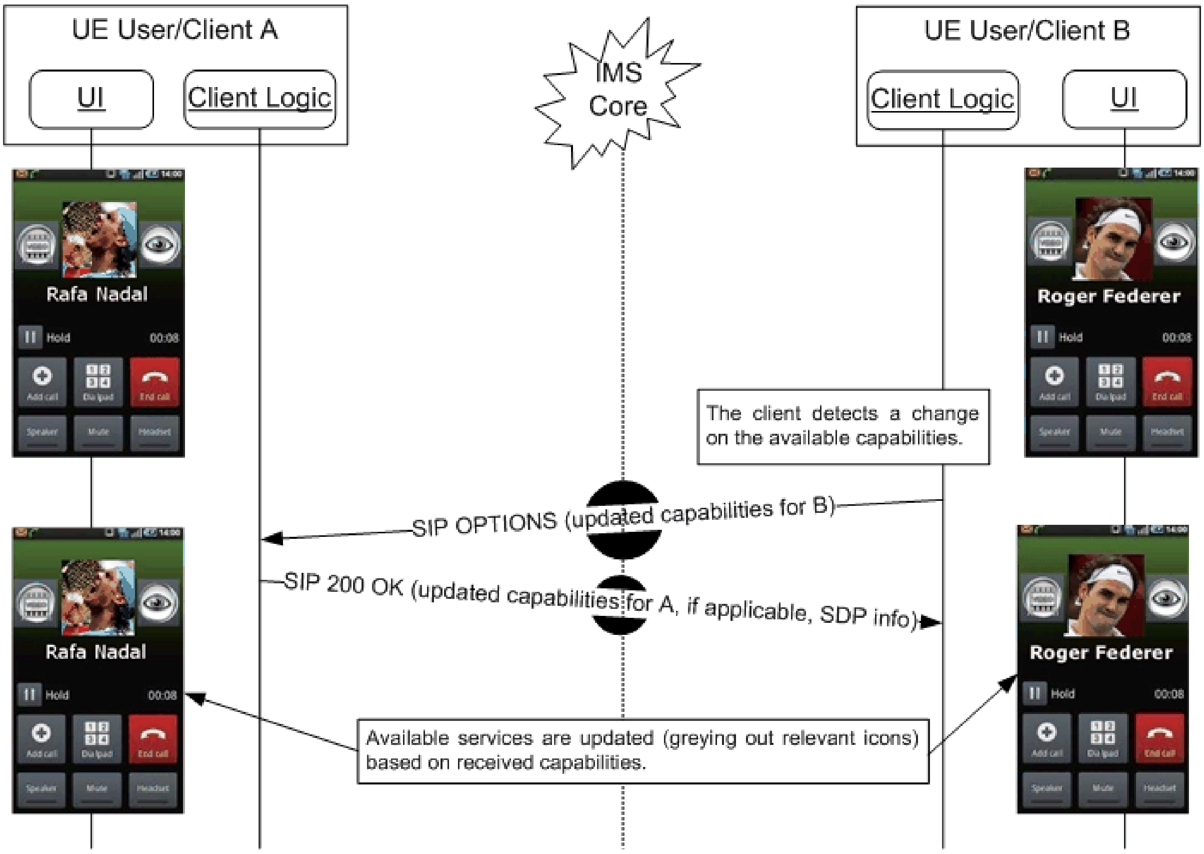

Figure 64: Capabilities exchange during a call

3.6.4.7.3 Share video

The assumptions in this case are that both User A (wanting to share video) and User B (recipient wanting to receive it), have successfully performed the capability query, as shown in section 3.6.4.7.2. Therefore, both clients are aware that video sharing is possible (both UEs on a 3G+ or Wi-Fi).

[37] The SDP information included in the response to the OPTIONS request is required due to the compliancy to [PRD-IR.74]. This will only be used during OPTIONS exchanges related to a call. The Video Share service shall only be considered to be available if at least one codec in the received SDP is supported by the client.

In this case, RTP is the protocol used to stream the video data, so it can be re-produced in real-time on the other end.

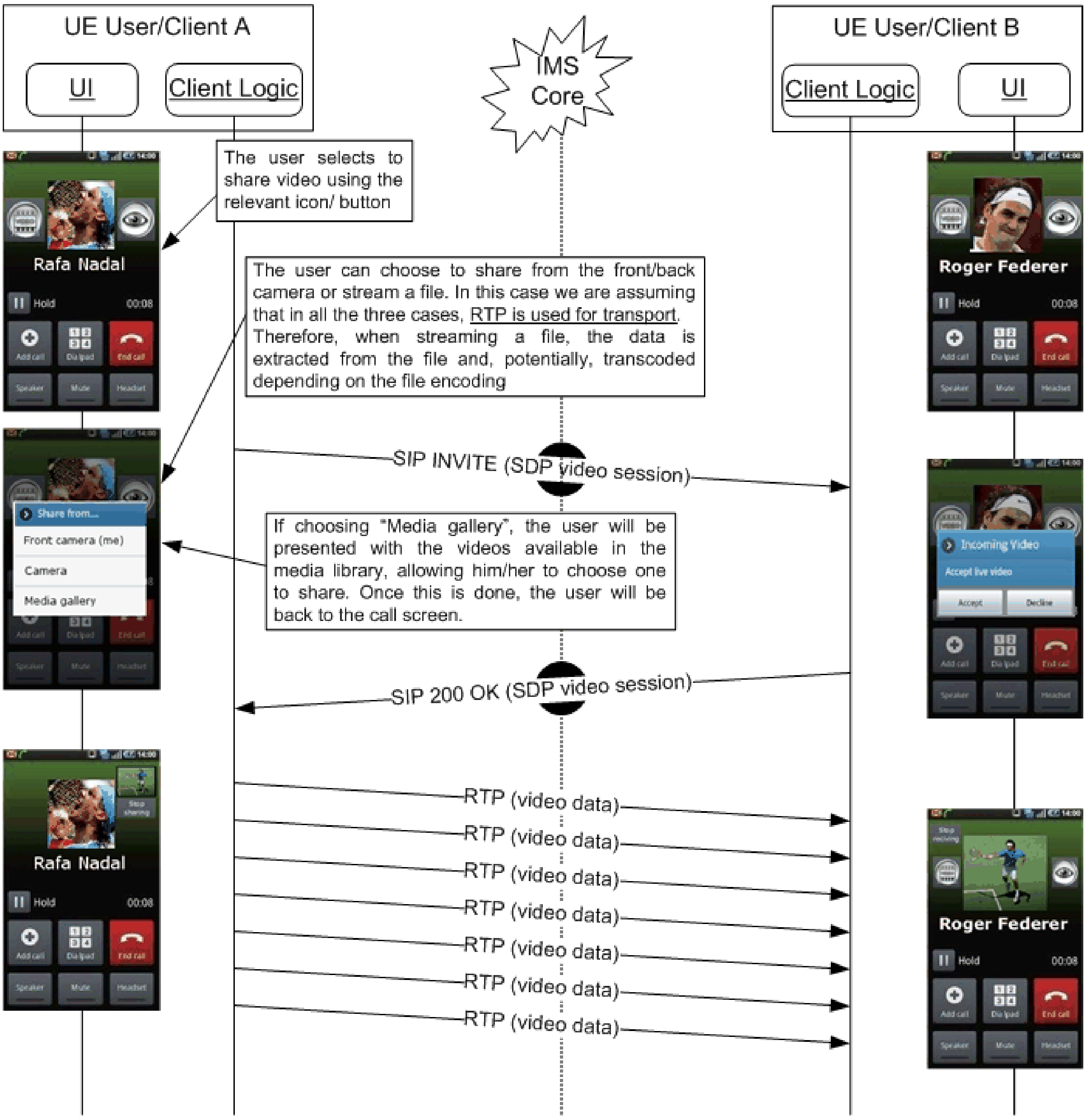

Figure 65: Share Video

3.6.4.7.4 Stop sharing video (RTP): Sender initiated

The assumptions in this case are that User A is sharing a video (through RTP) with User B. However, User A no longer wants to keep sharing it.

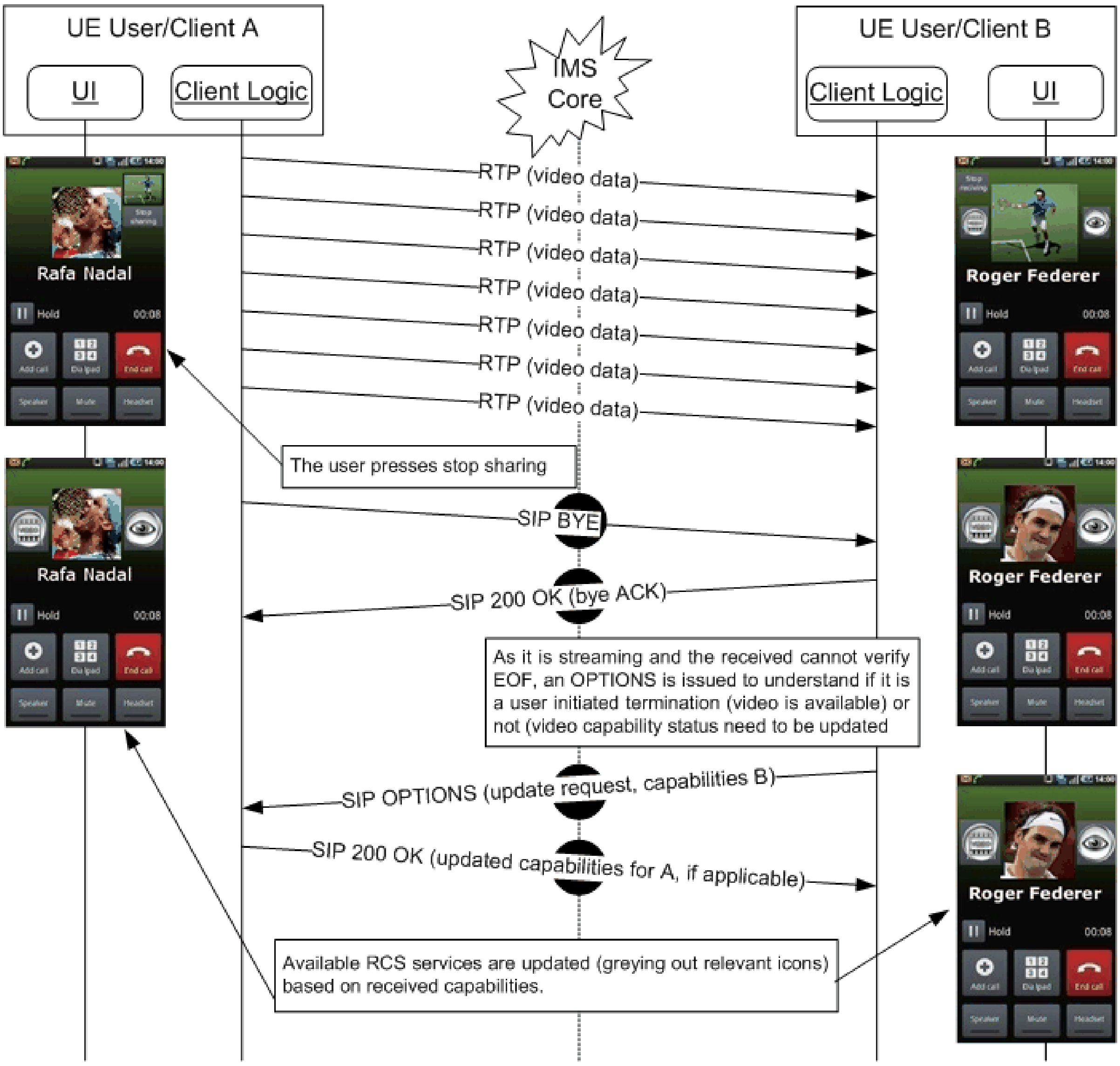

Figure 66: Sender stops sharing video

3.6.4.7.5 Stop sharing video (RTP): Receiver initiated

This case is equivalent to the previous one. However, it is the receiver (User B) who does not want to keep receiving the video.

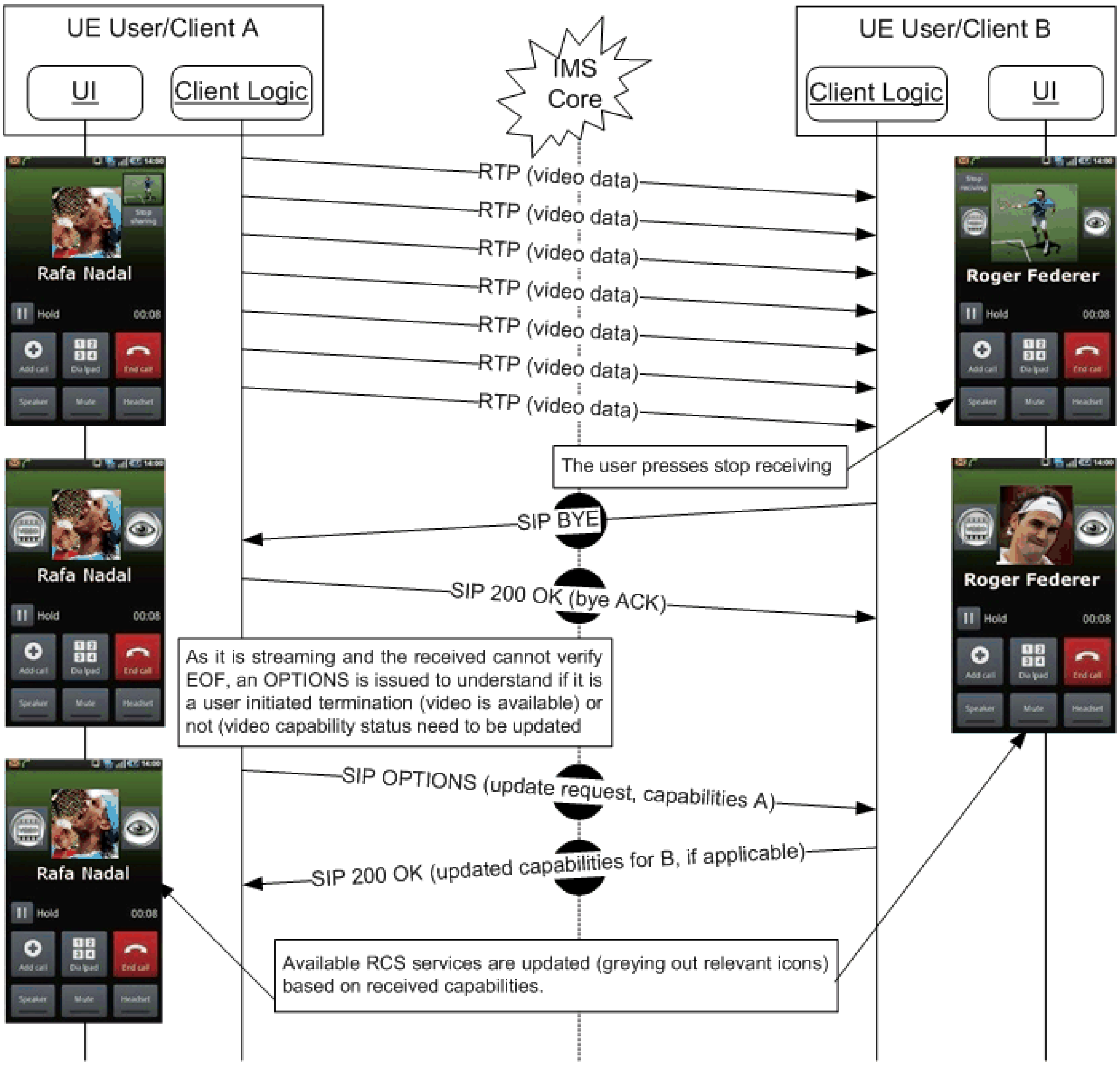

Figure 67: Receiver wants no longer to receive video

3.6.4.7.6 Stop sharing video (RTP) as the required capability is no longer available

The assumptions in this case are that User A is sharing video (RTP) with User B, and either User A or User B is no longer capable (for instance because the terminal is busy, suddenly has no LTE, 3G+ or Wi-Fi coverage available without triggering an IP reconfiguration or loss of connection) of sending or receiving a video. Please note that in the example, it is assumed that the sender (User A) is the one losing the capability. This sequence will be equivalent if:

- The receiver (User B) loses the capability to receive video: The BYE and OPTIONS exchange would be initiated by the receiver (User B) in this case.
- Both lose the capability to share video: The BYE and OPTIONS exchange message would be initiated by the client that is the first one to lose the capability in this case.

In losing the capability to send video, the case in which there is an IP reconfiguration is excluded. Please note that this particular case is covered under the "Client Error" section later in this section (see 3.6.4.7.12 and 3.6.4.7.13).

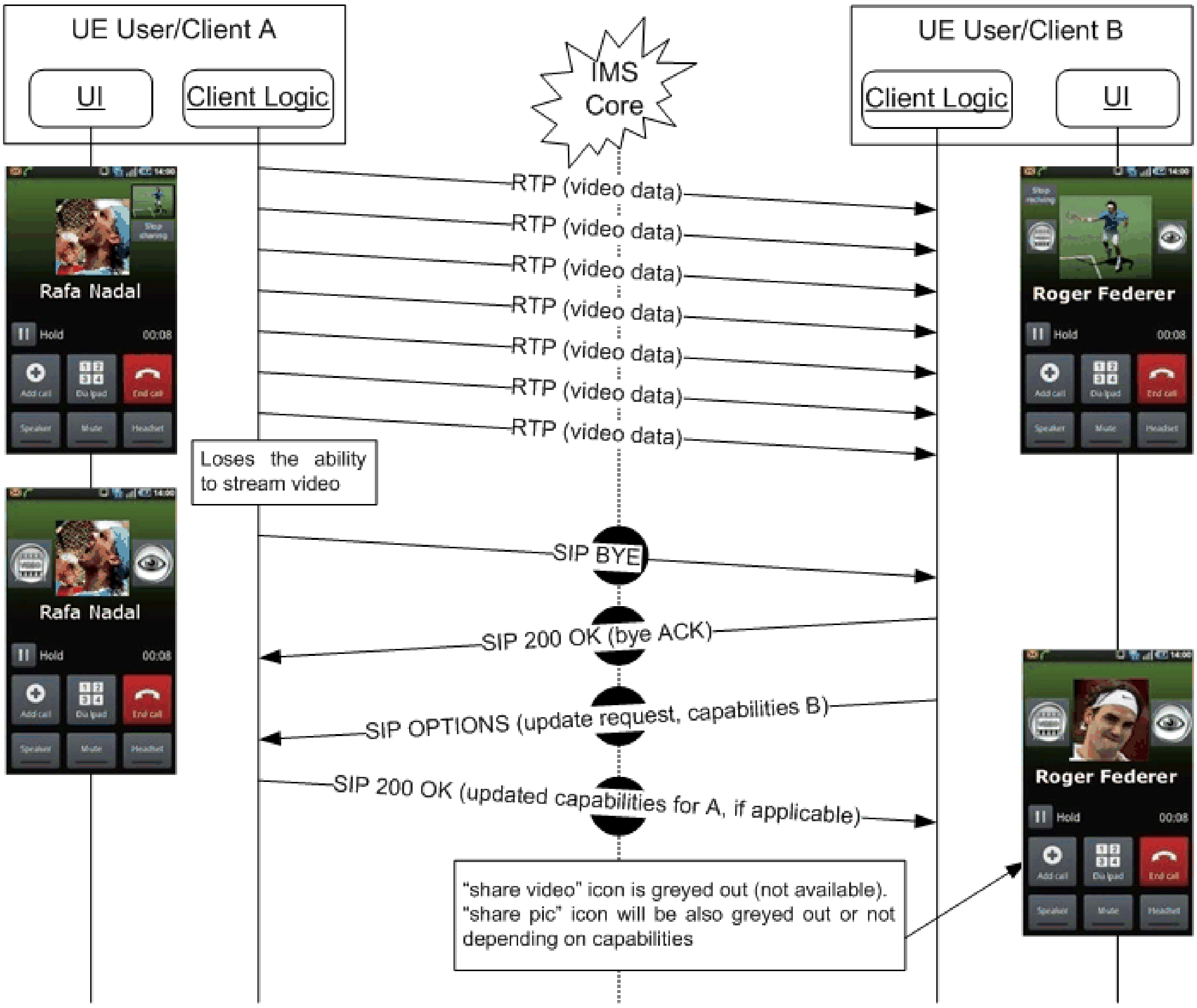

Figure 68: Video can no longer be shared (capability not available)

3.6.4.7.7 Share pictures during a call

The assumptions in this case are that both User A (wanting to share picture) and User B (recipient wanting to receive it), have successfully exchanged the OPTIONS messages. Therefore, both clients are aware that Image Share is possible (that is both UEs are on an LTE, 3G+ or Wi-Fi network).

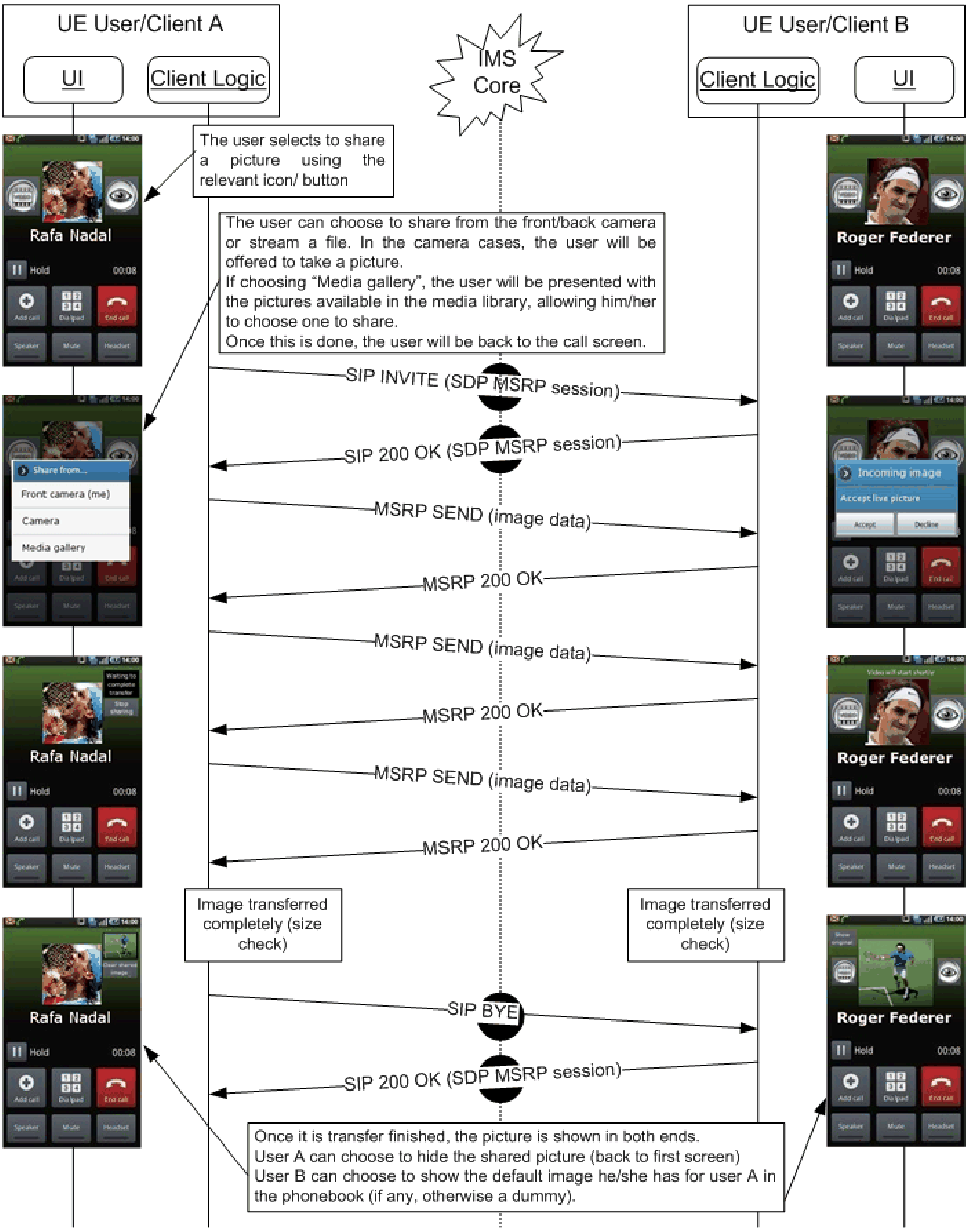

Figure 69: Sharing a picture during a call

3.6.4.7.8 Stop sharing a picture during a call: Sender initiated

The assumptions in this case are that User A is sharing a picture with User B, the transfer is still ongoing, but User A no longer wants to keep sharing the picture.

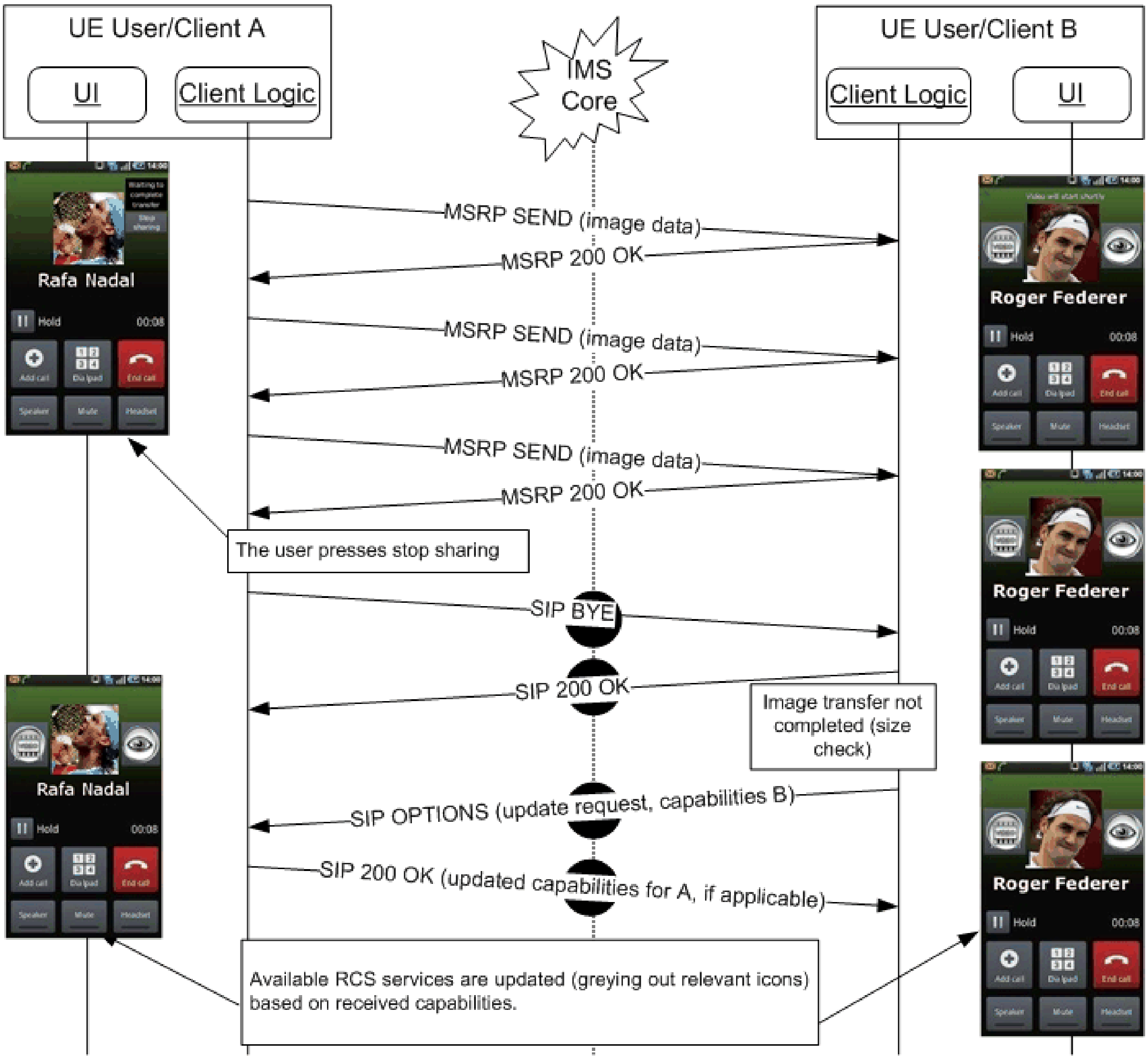

Figure 70: Sender stops sharing a picture during a call

3.6.4.7.9 Stop sharing a picture during a call: Receiver initiated

This case is equivalent to the previous one. It is, however, the receiver (User B) who does not want to keep receiving the picture.

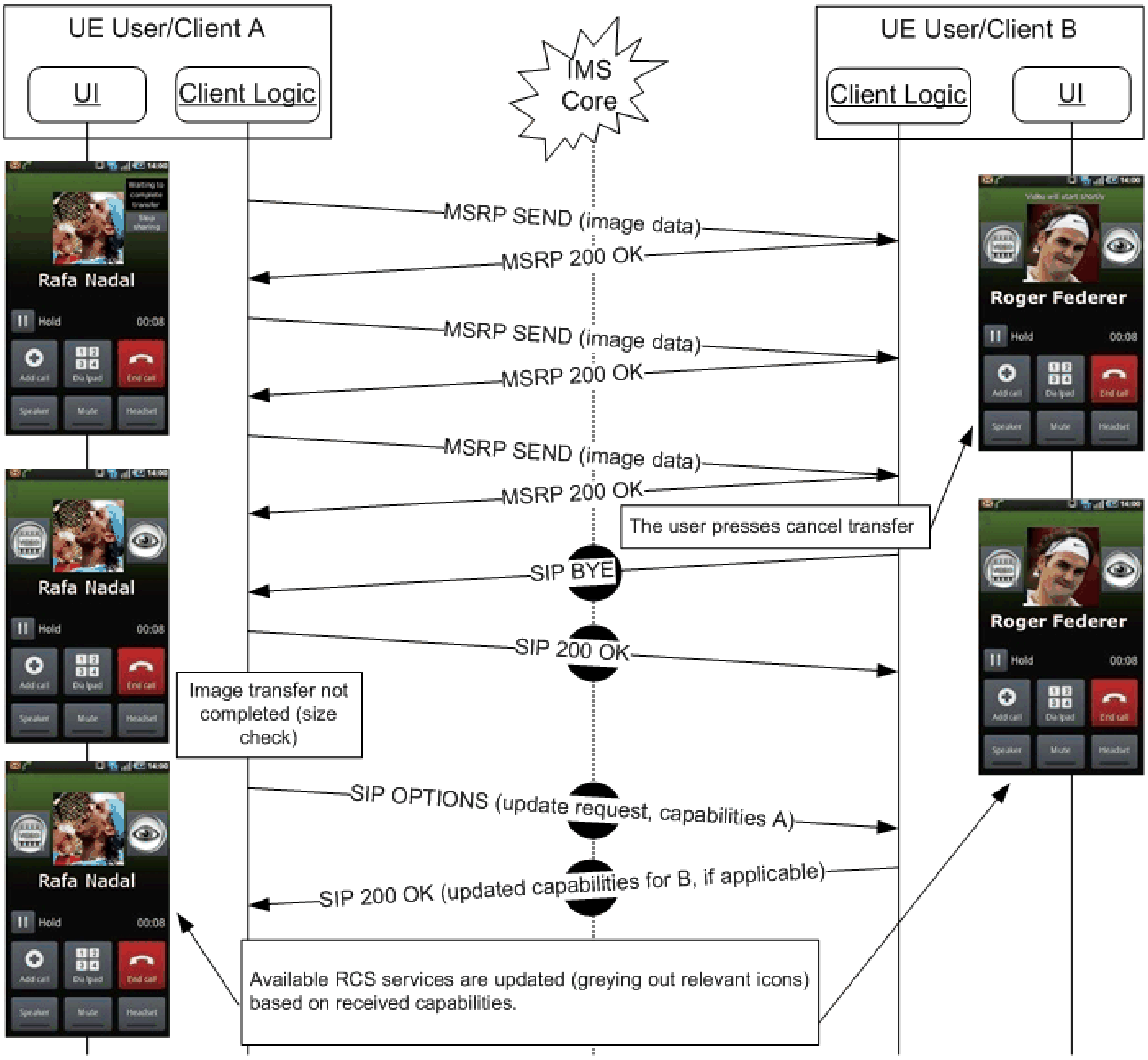

Figure 71: Receiver stops picture sharing

3.6.4.7.10 Stop sharing a picture during a call as the required capability is no longer available

The assumptions in this case are that User A is sharing a picture with User B, the transfer has not yet finished, and either User A or User B are no longer capable (for instance because the terminal is busy) to sharing or receiving the image respectively. Please note that in the example it is assumed that the sender (User A) is the client losing the capability. The sequence will be equivalent however for:

- The Receiver (User B) losing the capability to receive pictures: The BYE and OPTIONS exchange would be initiated by the receiving client (User B) in this case.
- Both lose the capability to share pictures: The BYE and OPTIONS exchange would be initiated by the first client to lose the capability in this case.

Please note that there is an exception to stop a file transfer due to capabilities. If one of the users is left with 2G coverage (on a DTM terminal) once a transfer has started, the transfer may continue until completed, provided the handover did not trigger an IP bearer reconfiguration. Once the transfer is completed however, picture sharing will no longer be available as a service during the call.

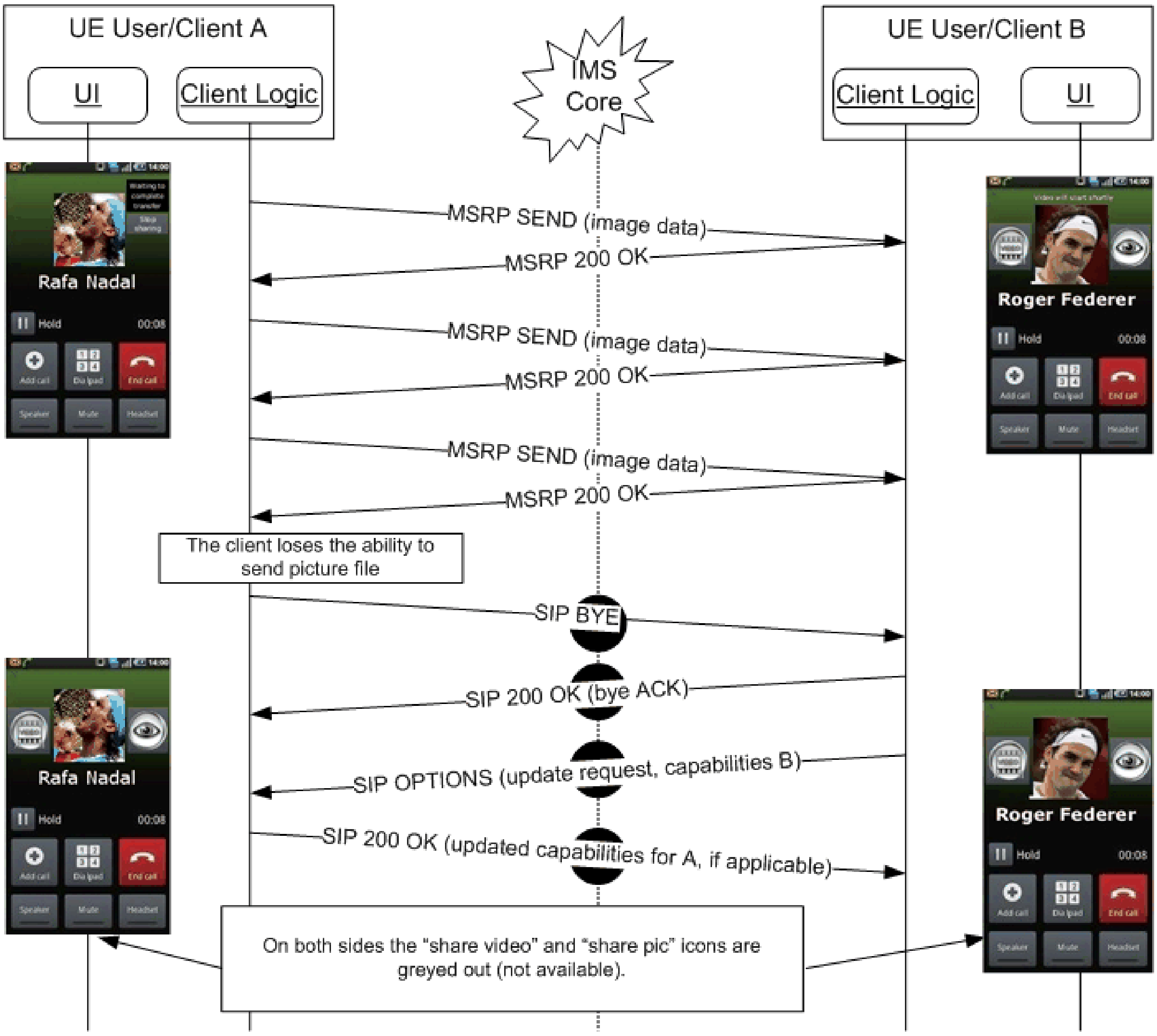

Figure 72: A picture can no longer be shared during a call (capability not available)

3.6.4.7.11 Decline share video or picture

User A wants to share a video or picture with User B. User B, however, does not want to receive it. Please note that it is assumed that both Video and Image Share is possible (that is the proper capabilities are available).

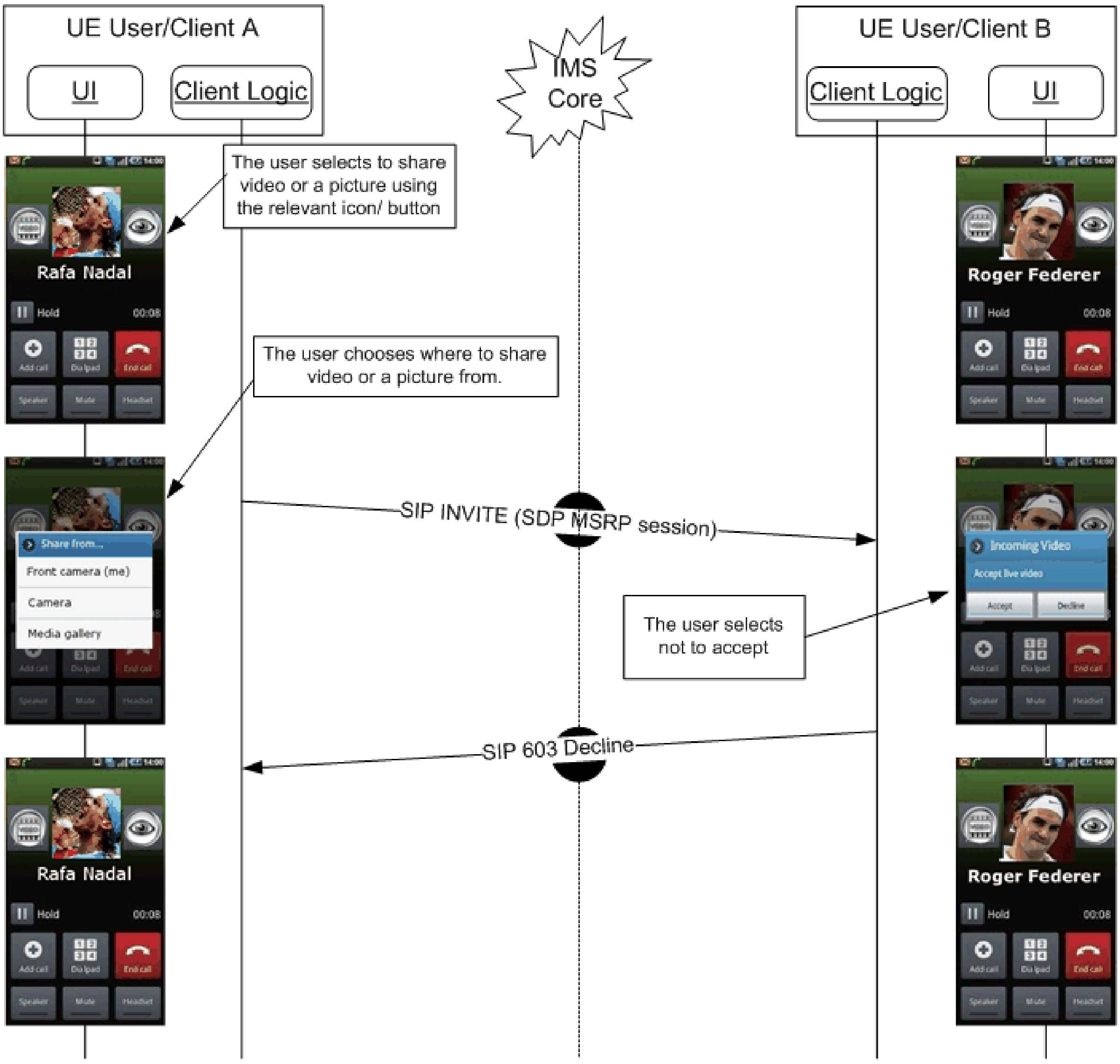

Figure 73: User declines sharing a picture during a call

3.6.4.7.12 Non-graceful termination (sender): Video or picture sharing

In this case, User A is sharing video or a picture with User B. Suddenly, User A's connection to the network fails (This may for instance be due to a client error, a reboot of the device, the loss of the data bearer, a switch in data carrier [for instance 3G+ to 3G] causes an IP layer reconfiguration and so on).

In the following flow, it is assumed a video transfer (RTP) was taking place. It will be equivalent however to the case an MSRP transfer (Image Share or video sharing via File Transfer) was taking place and was not finished:

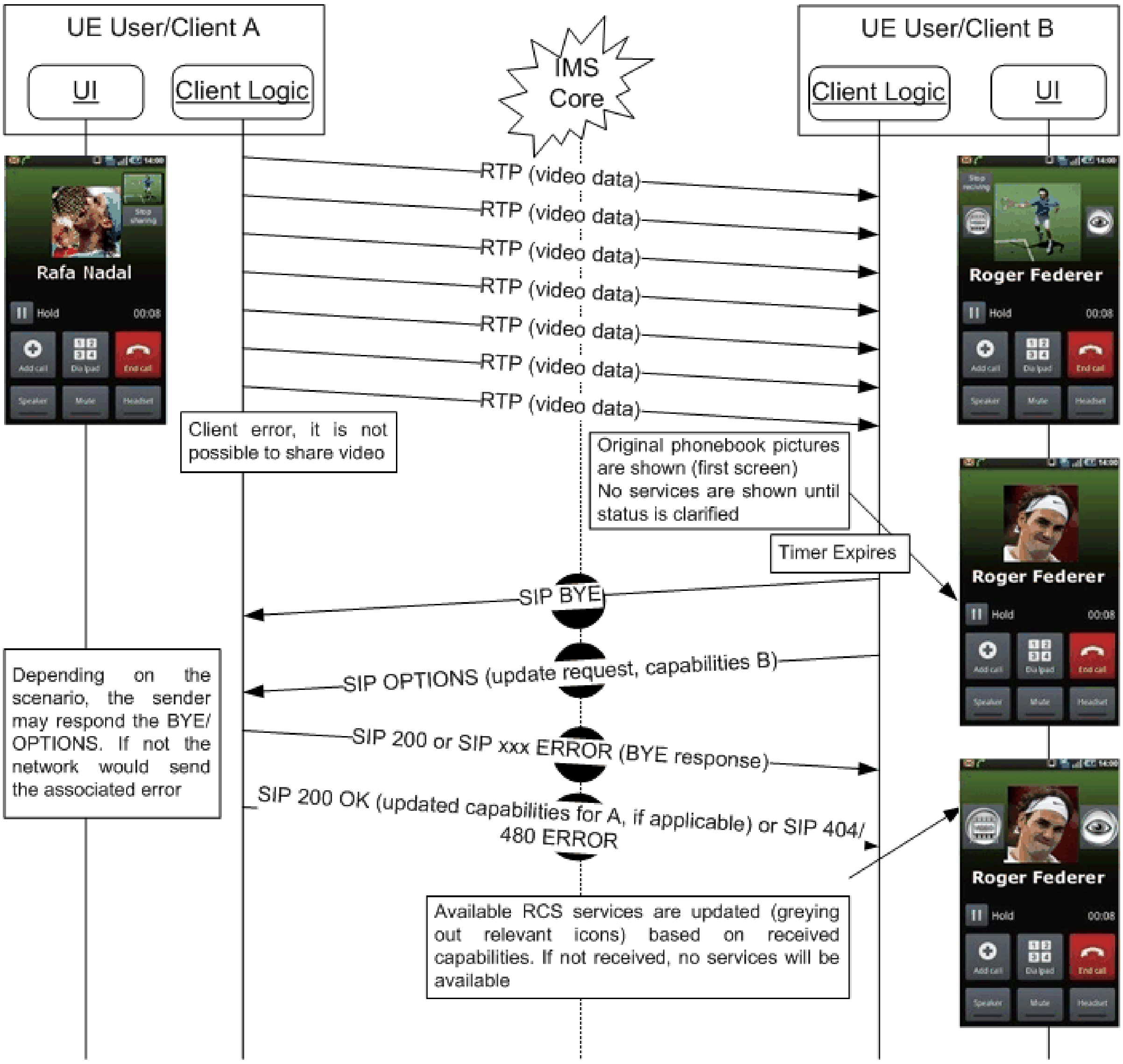

Figure 74: Non-graceful termination (sender) for video

3.6.4.7.13 Non-graceful termination (receiver): Video or picture sharing

To protect the IMS Core network from cases where both the sender and the receiver become unresponsive or unreachable before they had time to terminate the SIP session, the RCS Client shall use the procedure described in [RFC4028] in a similar way to the one mandated in [RCS-SIMPLEIM-ENDORS], that is the RCS client initiating a SIP session must request the role of refresher and the option tag 'timer' must be included in a Supported header.

The Session-Expires and Min-SE values announced by an RCS client must be configurable by the Service Provider.

This use case is identical to the previous use case, except that in this instance User B (receiver) loses the ability to receive/process MSRP messages (this can for example be due to a client error, a reboot of the device, a loss of the data bearer and so on).

In the first flow diagram it is assumed that an Image Share transaction was taking place through MSRP:

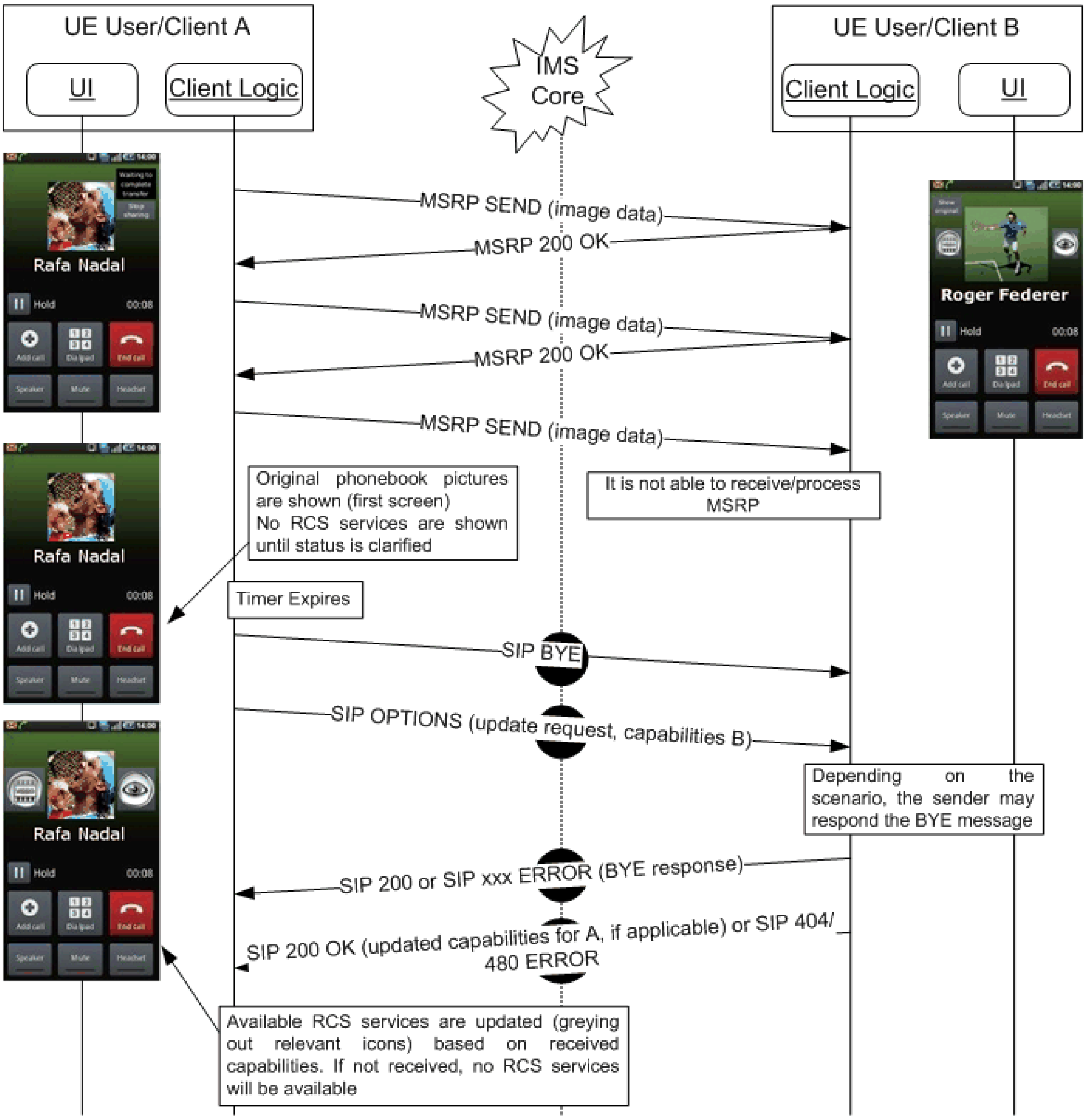

Figure 75: Non-graceful termination of picture sharing during a call

In the second flow it is assumed that a Video Share transaction was taking place through RTP:

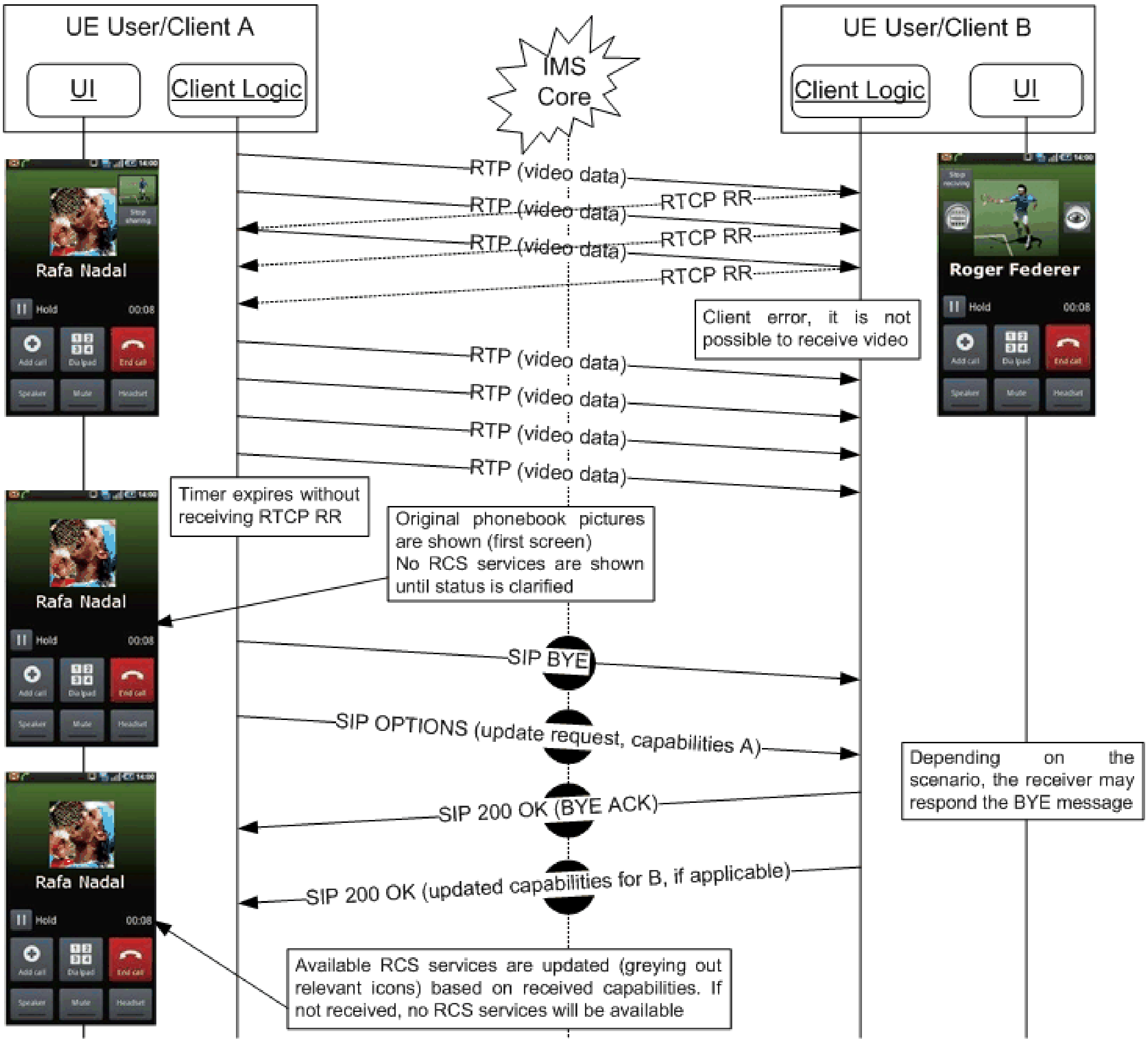

Figure 76: Non-graceful termination of video sharing during a call

3.6.4.8 Call Composer flows

Flows related to Call Composer service are provided in Annex B of [PRD-RCC.20].

3.6.5 NNI and IOT considerations

The NNI interfaces for content sharing services shall behave according to the procedures described in section 2.12 and referred documents.

3.6.6 Implementation guidelines and examples

3.6.6.1 Image, video, map or drawing canvas sharing during a call

As this is about sharing during a call, for both the sender and the receiver the sharing always starts from the call screen where the capabilities for sharing to the conversation partner in the voice call are shown. The user can then select one of the available services after which they will select the source of the sharing. A session will then be set up and the user will see the content that is being shared.

3.6.6.1.1 Video Share

The description above leads to following user experience for the initiator of a Video Share:

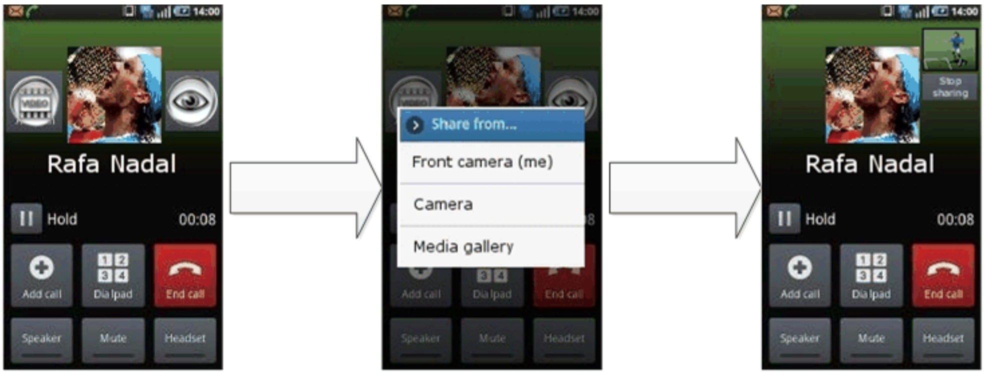

Figure 77: Reference UX for Video Share during a call (initiator)

A user invited for Video Share during a call first receives an additional invitation and if they accept, they are shown the video with the possibility to stop the sharing:

Figure 78: Reference UX for Video Share during a call (recipient)

NOTE: When the receiver accepts the sharing from the device that is involved in the voice call this acceptance applies automatically to all further sharing requests during that call.

3.6.6.1.2 Image Share

For Image Share, the experience is similar than the one for Video Share shown in section 3.6.6.1.1. As it requires the transfer of a large file before something can be displayed rather than being able to stream immediately, there is a transfer delay. This leads to the following user experience for the sender:

Figure 79: Reference UX for Image Share during a call (sender)

A user invited for Image Share during a call first receives an additional invitation and if they accept, they are shown the image with the possibility to stop the transfer initially and stop displaying the image once transferred:

Figure 80: Reference UX for Image Share during a call (receiver)

NOTE: When the receiver accepts the sharing from the device that is involved in the voice call this acceptance applies automatically to all further sharing requests during that call.

3.7 Social Presence Information

3.7.1 Feature description

3.7.1.1 Social Presence definition

Social presence is seen as a piece of information for buddies to let them know about what you are doing, your mood, status, and so on. The user is given the possibility to publish personal data, which configures the users Social Presence Information, or “personal profile”.

As an illustration, the group of contacts with whom a presence relationship is established can be seen as the closest contacts of a certain user (friends, family, colleagues, and so on.).

Social Presence Information (included in the personal profile) does not replace the legacy contact’s vCard in the address book of the user (for example. the contact name and other contact details shall not be impacted).

The Social Presence Information shall be controlled by the end user and easily configurable.

Having established a Social Presence Relationship with a certain contact, the Social Presence Information shall be visible from the Enhanced Address Book (EAB). It should also be visible from other places on the device, like for example the communications log, or message folders.

3.7.1.2 Service Fundamentals

In the EAB, the contact information is extended with social presence information and foresees the following attributes:

- Availability, indicates the user’s (un)willingness to communicate,
- Portrait icon, depicting the user (e.g. a photo or image provided by the contact himself)

- Free text, including textual note and possibility to add emoticons (automatic translation of some specific characters into smileys)
- Favourite link, to publish hypertext link of personal and/or favourite site
- Timestamp, date of the last update of the profile, generated automatically.
- Geolocation, depicts the user location

The attributes Availability, Portrait icon and Favourite link are profiled from the standards bringing a new user experience.

The Availability allows a user to inform a contact that they are currently in a situation when it is possible/not possible to communicate.

The Availability is controlled fully by the user and not automatically switched on or off.

With the portrait icon, it is possible to publish a photo or an icon, which is shown in the EAB of the user's contacts. This is a new user experience while a user has full control of the portrait displayed at his contacts. Within RCS the size and dimension of the photo is specified.

The favourite link attribute allows sharing additional social presence information. Such a link can point to e.g. a blog.

With geolocation, two RCS users are able to see where they are located and share this information with each other.

Authorisation to share social presence is based on the symmetry principle.

If sharing of social presence is accepted after invitation, both parties will see each other's presence attributes. If social presence sharing is terminated by one of both parties, both parties will end seeing each other's social presence attributes.

When a social presence relationship with a contact is set up from one device (e.g. the broadband client on PC) this relationship will also be visible on the other devices of the user (e.g. a mobile device).

The RCS invitation experience is improved with a personalized invitation. For easy identification of invitations coming from contacts not yet registered in the user's address book, it is possible to define a nickname to be used in presence invitations.

By choosing whether or not the contact is a VIP contact (see section 3.7.1.4.9), it will be possible to choose for a contact with which social presence is shared whether updates to that contact's social presence information should be reflected in (near) real time or whether those updates should be retrieved through some low frequency polling for them.

3.7.1.3 Social presence attributes

3.7.1.3.1 Availability status

A user will be able to set the state of Availability status (as part of Social Presence Information)

There are two possible states that can be selected by the user, from their RCS Client:

1. **State#1.** From the RCS Client, the user can set Availability status information as state#1. This state is informative and means that user is available and willing to communicate The way state#1 is displayed to the user is implementation dependent, and subject to own Service Provider policies.
2. **State#2.** From the RCS Client, the user can set Availability status information as state#2. This state is informative and means that the user is unavailable or not willing to communicate (e.g. busy) and will probably not respond to any incoming calls or messages. The way state#2 is displayed to the user is implementation dependent, and subject to own service provider policies.

These states are informative. When a user sets Availability status information as state#1 or state#2 from the RCS Client, the user still has the possibility to make outbound communications (e.g. calls/messages) and receive inbound communications (e.g. calls/messages).

The Availability status information has a permanent nature. It remains unchanged until the user decides to modify it (as state#1 or state#2) from their RCS client

The Availability status information is not linked with any particular user's network connectivity situation (e.g. temporary loss of network connectivity, device switched off).

The RCS device and the Presence Server shall support the availability status feature.

3.7.1.3.2 Favourite Link

One of the attributes in the Social Presence Information allows the user to add or update one hypertext link, which (when selected) may redirect, for instance, to an extension of the user's Social Presence Information (for example a mobile blog).

The user shall be able to edit the hypertext link (expressed as a Uniform Resource Identifier as defined in [RFC2396]).

A clickable link is displayed in a *detailed view mode* of the Social Presence Information, where shared information about the user (portrait icon, free text and URI) can be seen in larger size than in the EAB itself (*list mode*).

When the user edits a new hypertext link, those contacts, which the user has established a Social Presence Relationship with, are notified, that is a visual change of value of favourite link attribute, for example when the user updates their portrait icon or free text.

When a user clicks on the link of a presence-enriched contact, the appropriate native handler for linked content (for example browser) shall be launched.

When the user closes the handler, they return automatically to the presence enriched contact's *detailed view mode* of the Social Presence Information, from where the handler was launched.

A revoked contact shall not be able to click on the hypertext link. However, please note that there are no restrictions that prevent the watcher from being able to save the URI in their browser and further access to this URI.

It is possible to display a "user friendly" label for the favourite link instead of the actual URI.

Instead of displaying the URI the RCS user can display a personal label. The maximum size of characters is 200 characters.

3.7.1.3.3 Geolocation information

Geolocation information is a combination of declarative text always manually edited/updated by the user; and/or coordinate information (x, y) that is displayed on a map.

The maximum character size of declarative location text information the end-user can enter shall not exceed 200 characters.

Time Zones can be shared as part of geolocation information, allowing users to view what the local time is at their friend's location.

A provisioning parameter can be set in the network by Service Providers to control the maximum time the published location information will be considered to be valid (for example, one month).

The user must be able to delete his location information (empty text field, no position on map).

Location information must be interoperable between RCS clients no matter how users choose to update their information. For example, if User A has updated his location on a map (with x, y coordinates) and User B (authorised contact) is using RCS clients without a map feature (and only supporting declarative text), they must still be able to view User A's location as a intelligible text, using the declarative text information (if available), not as raw x, y information.

To avoid excessive traffic on the network due to very frequent location updates, it is recommended that a provisioning parameter can be set in the network to remotely set a minimum duration between updates sent from the client/device.

The geolocation feature can be provided on non-GPS (Global Positioning System) enabled devices.

3.7.1.4 Social Presence Authorisation

RCS users shall feel confident in publishing their Social Presence Information, and be guaranteed that their privacy is respected. Therefore, mechanisms are defined below that allow users to accept/reject an invitation to establish a Social Presence Relationship, since this may imply sharing certain potentially private information, such as portrait icon or free text.

3.7.1.4.1 Social Presence Information sharing request principles

Reactive authorisation shall be used, that is when User A invites User B to share Social Presence Information, User B receives an authorisation request.

When receiving an invitation to share Social Presence Information from User A, User B can:

- **Accept** the invitation.
- **Ignore** the invitation, which requires an explicit action by User B.
- **Block** User A from sending more invitations.

- **Not answer**, that is do nothing with that request.

Invitation to share Social Presence Information automatically implies the authorisation of the requesting user, that is, when User A invites User B to share Social Presence Information, User A automatically authorises User B to see their Social Presence Information.

If User A's MSISDN is associated with a contact in User B's address book, the name given to that contact shall be displayed within the invitation to share Social Presence Information.

Symmetric authorisation shall be used. The publication of Social Presence Information shall be bidirectional.

User A shall not receive any notification whether User B has not answered, blocked or ignored their invitation to share Social Presence Information.

Once a Social Presence Relationship has been established, the user can stop that relationship via the following action:

- **Revoke** the Social Presence Relationship.

3.7.1.4.2 Accept

If User B accepts User A's invitation to share Social Presence Information, User A will see User B's Social Presence Information, and User B will see User A's Social Presence Information.

If User A is not an existing contact in User B's address book, it shall be facilitated that User B stores the contact details of User A in their address book.

3.7.1.4.3 Ignore

If User B ignores User A's invitation to share Social Presence Information, neither User A nor User B shall be able to see each other's Social Presence Information.

Ignoring an invitation to share Social Presence Information shall not mean blocking the contact that has sent the invitation, i.e. it shall still be possible to receive more invitations from that contact.

If User B ignores User A's invitation to share Social Presence Information but later, User B decides to share their Social Presence Information with User A, then it is not necessary that a new authorisation request is issued to User A. User B, by adding User A to their EAB completes the symmetric authorisation process. As a result, User A and User B will be seeing each other's Social Presence Information.

3.7.1.4.4 Block (refuse to receive any further invitation)

In order not to receive more invitations from a certain contact, the user shall be given the possibility to add that contact to a list of blocked contacts (blacklist).

The blocking mechanism shall be transparent to the blocked user, that is, if User B blocks User A, User A shall never be notified that he/she has been blocked by User B.

The possibility shall be given to remove a certain contact from the blacklist, i.e. User B shall be able to see in their EAB that User A has been blocked and to remove them from the blacklist.

3.7.1.4.5 Not answer (pending invitation)

If User B does not answer User A’s invitation to share Social Presence Information, the invitation shall be in a pending state, for which an action is expected by User B.

Pending invitations to share Social Presence Information with User A, for which an answer has not yet been provided, shall be accessible for User B, so that User B can choose to answer the invitation that is Accept, Ignore or Block.

Subsequent invitations (from User A to User B) replaces User A's initial invite, and function as a reminder for User B that a corresponding action (on their part) regarding the invitation to share SPI is required. That is, User B needs to choose an option

- Accept,
- Ignore, or
- Block.

3.7.1.4.6 Revoke

Once a Social Presence Relationship has been established, the possibility shall be given to stop the sharing of Social Presence Information with a certain contact, while at the same time removing your Social Presence Information from that contact’s EAB.

If User A revokes the Social Presence Relationship with User B, both users shall not receive any further updates of their Social Presence Information, according to the symmetry principle.

When User A revokes the Social Presence Relationship with User B, User B shall no longer be displayed as a presence enriched contact.

User B’s Social Presence Information shall not be shown to User A

Only User B’s contact details (vCard) shall remain visible in User A’s address book (for example name, MSISDN, e-mail, and so on.

If User A revokes the Social Presence Relationship with User B, User B shall no longer have access to User A’s Social Presence Information.

Before actually performing the revoke, User A shall see a notification alert in the client informing him about consequences of this action. These are:

- User A’s Social Presence Information will be removed from User B’s EAB, so User B will notice the revoke after a certain period of time (for example several hours/days)
- It will be possible for User A and User B to invite each other again.

After a Social Presence Relationship has been revoked for a given period of time (for example several hours/days), both users can reinitiate the process of Social Presence Authorisation, that is User A shall be able to invite User B to share Social Presence Information and vice versa.

It must be noted that User A may immediately re-invite User B to share Social Presence Information.

If User A deletes User B's vCard from their address book, all contact information is deleted from User A's address book. If a Social Presence Relationship between User A and User B exists at the moment of deleting the contact, this relationship shall be revoked.

3.7.1.4.7 Personalised Invitation

To improve RCS invitation experience with a personalised invitation and to ease identification of invitations coming from contacts not yet registered in the user's address book, a nickname feature is provided:

- If the terminal supports configuring a nickname, the user can choose a "nickname" with limited size (recommendation: 20 characters, this size can be set as a provisioning parameter). This nickname is provided in all future invitations to share presence, until it is changed. The maximum number of characters an invitee can view is 200 (this limitation is proposed to ensure interoperability for invitee, regardless of the number of characters implemented by the service provider).
- The invitee, if they do not have the inviter information in their address book, can now see both MSISDN and the nickname of the inviter.
- The nickname is stored permanently to be used for every invitation. Users have the ability to change it every time they send an invitation.
- The nickname does not replace the registered name of a contact already present in the recipient's address book.

Security: it is noted that through the use of the nickname, it is possible to "impersonate" someone. However, that "impersonation" is limited in scope since the inviting user remains identified by his MSISDN and the fact that the feature is only used for MSISDNs that are not already stored in the recipient's address book.

3.7.1.4.8 Geolocation authorisation

Two users should be able to see where they are located and share this information with each other and they would keep the control over this information:

- No specific invitation process for location.
- When and if a user chooses (by opt-in) to update their location for the first time, by default, users do not share their location information with all their contacts authorised for social presence.
- Users have the ability to manually choose contacts with whom they wish to share location information.
- Even if a user is not sharing location information with one of their authorised contacts, that does not prevent them from viewing that contact's location information.

Once User A has accepted User B as an RCS authorised contact, User B will be able to see the geolocation information of User A (displayed with a text or a map, or both of them) and all updates of that information.

When a given RCS user (User A) is willing to share Social Presence with another user, User A shall be able to control in the invitation process for sharing Social Presence whether sharing of their location information with this other user is authorised or not.

3.7.1.4.9 VIP contacts

As the number of SPI enabled contacts increases in the user's address book, the amount of information that the user receives in the mobile phone will increase making it more difficult to differentiate useful information from noise. In addition, the RCS users will not want to share the same Social Presence Information with all their contacts.

The selection of certain contacts as Very Important Person (VIP) contacts will allow the end user to specify which contacts are the most important ones.

The user should be able to differentiate the contacts, which they share SPI with, between important and unimportant contact. The user shall receive real time notification of status changes from VIP contacts.

The user will be able to choose from the contacts to set them as VIP contacts. The user will then only receive real time notifications of the social presence information from the contacts set as VIP contacts (probably with a phone buzz or light, or via an idle screen widget and so on). The contacts that are not set as VIP contacts will still be updated in the EAB, but not in real time, therefore, the user is made aware of the new social presence information when they browse the EAB.

3.7.1.5 Example Use Cases

3.7.1.5.1 Social Presence Information Use Cases

3.7.1.5.1.1 Invite Contacts to Share Social Presence

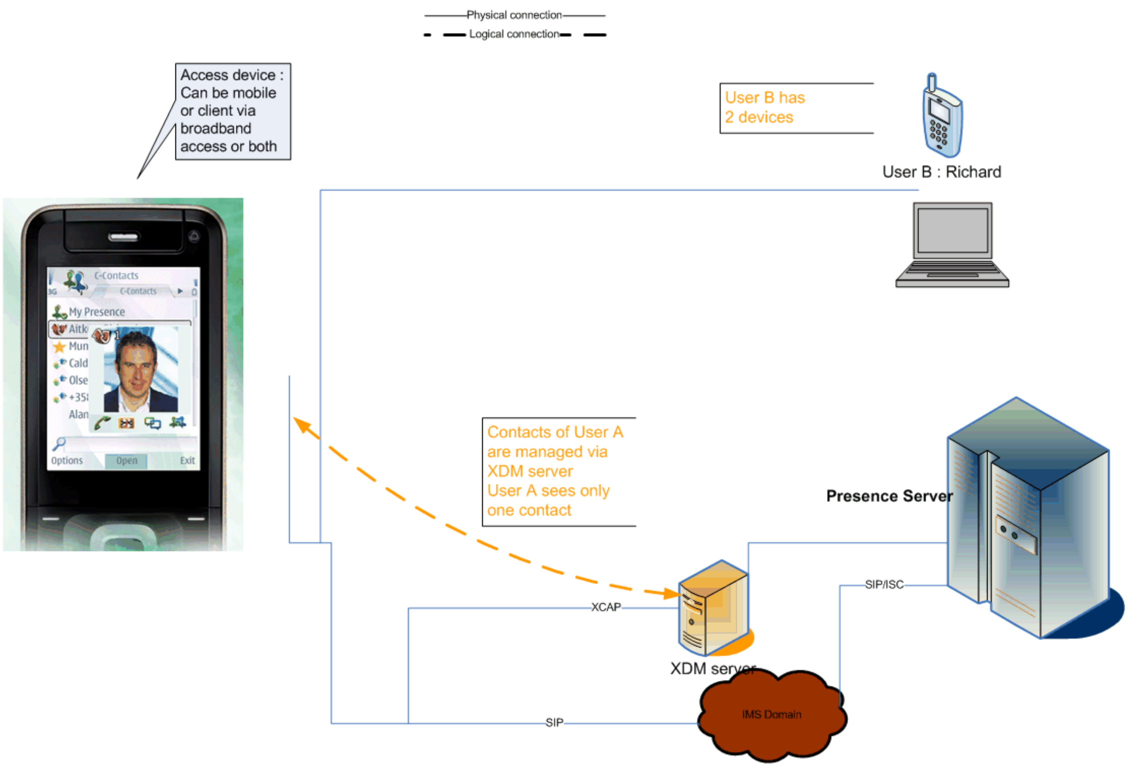

Figure 81: Invite Contacts to Share Social Presence

Authorisation to share social presence is based on the symmetry principle. If sharing of social presence is accepted after invitation, both parties will see each other's presence attributes. If social presence sharing is terminated by one of both parties, both parties will end seeing each other's social presence attributes.

It is possible to share with an invitation for social presence a nickname if the invited party does not have the inviting party's phone number in the device.

3.7.1.5.1.2 Allow Contacts to obtain Location Information

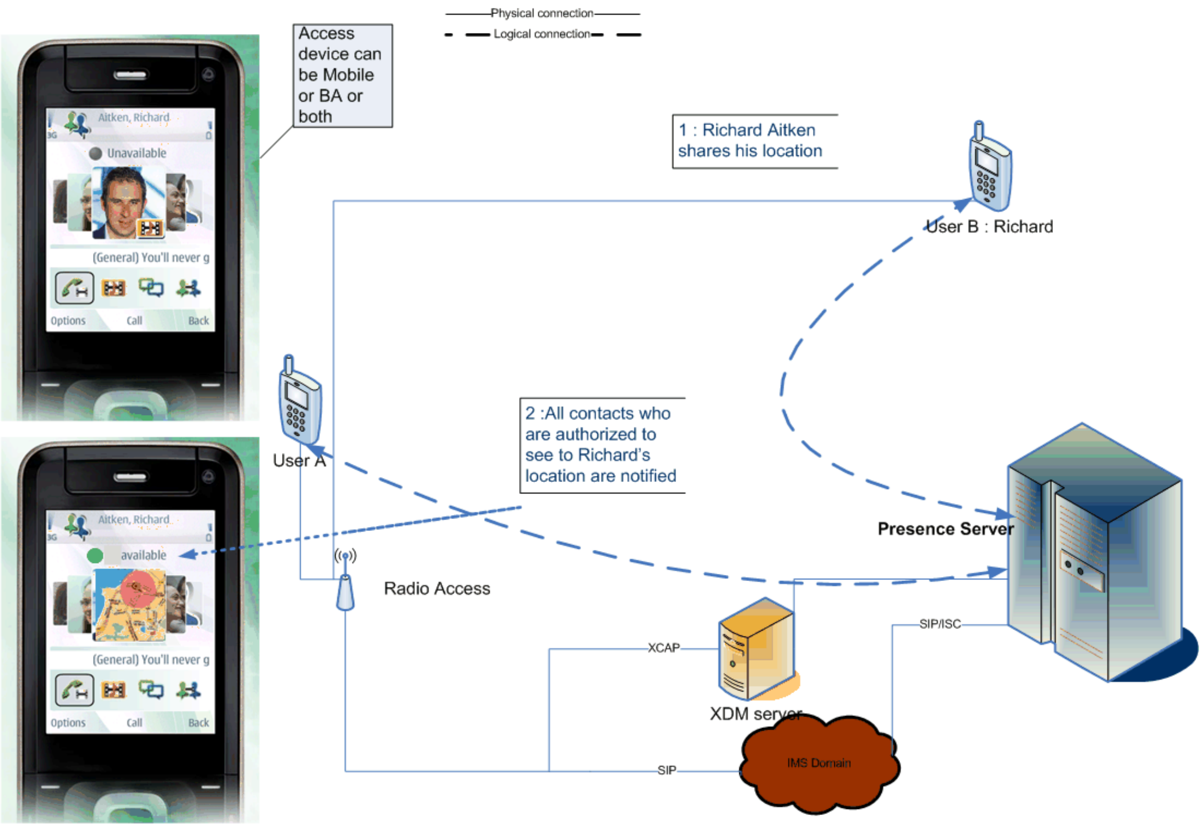

Figure 82: Share Location

This service allows users to show where they are through the RCS EAB and view where their friends are as free text and/or on a map.

NOTE: If the contact that is updated but not in the VIP group, the information (Richard Aitken in the use case) in their VIP group may not be seen immediately. They will only see it when either their client polls for updates of the non-VIP contacts or when they request for an update of the non-VIP contacts themselves. The user may even miss the update altogether if there is another update before the status of the non-VIP contacts is retrieved.

3.7.1.5.1.3 Availability

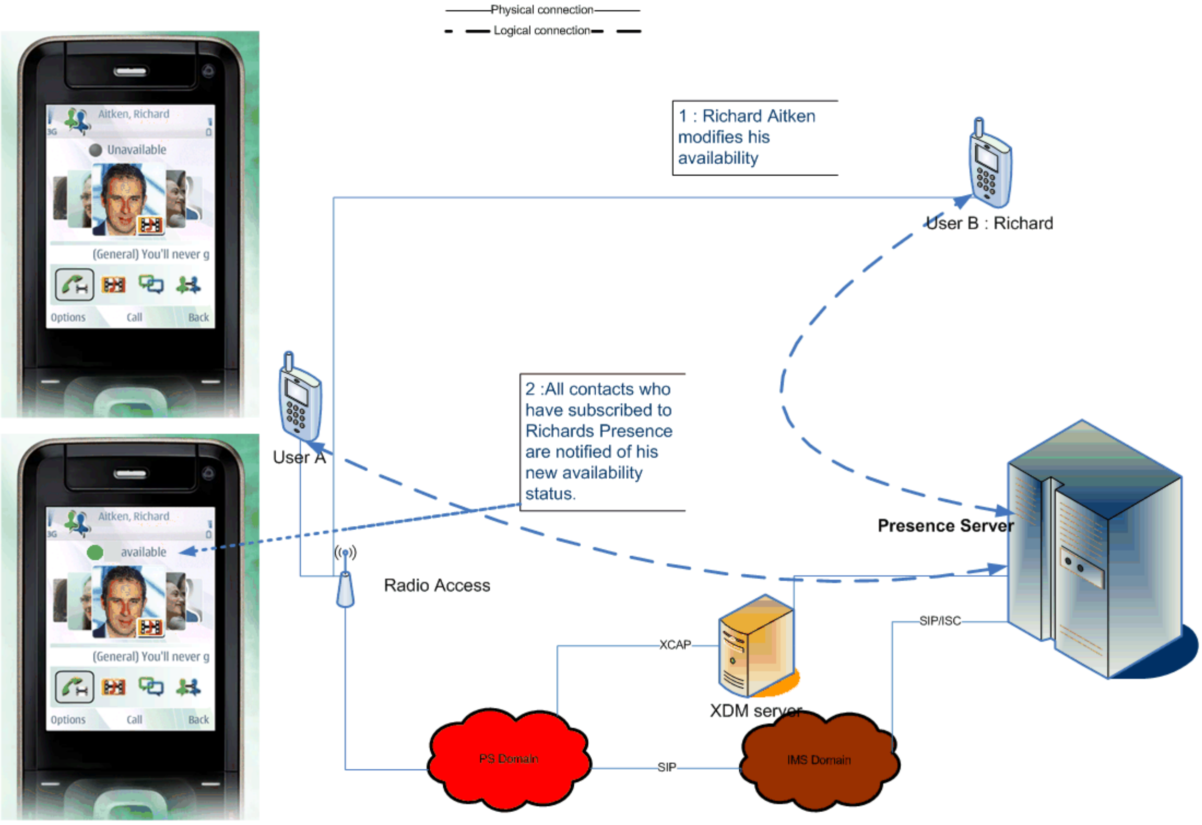

Figure 83: Availability

NOTE: If the contact that is updated is not part of the VIP group of the user the updated SPI (Richard Aitken's in the use case) may not be seen immediately. They will only see it when either their client polls for updates of the non-VIP contacts or when they request for an update of the non-VIP contacts themselves. The user may even miss the update altogether if there is another update of the availability status before the status of the non-VIP contacts is retrieved.

3.7.1.5.1.4 Free Text

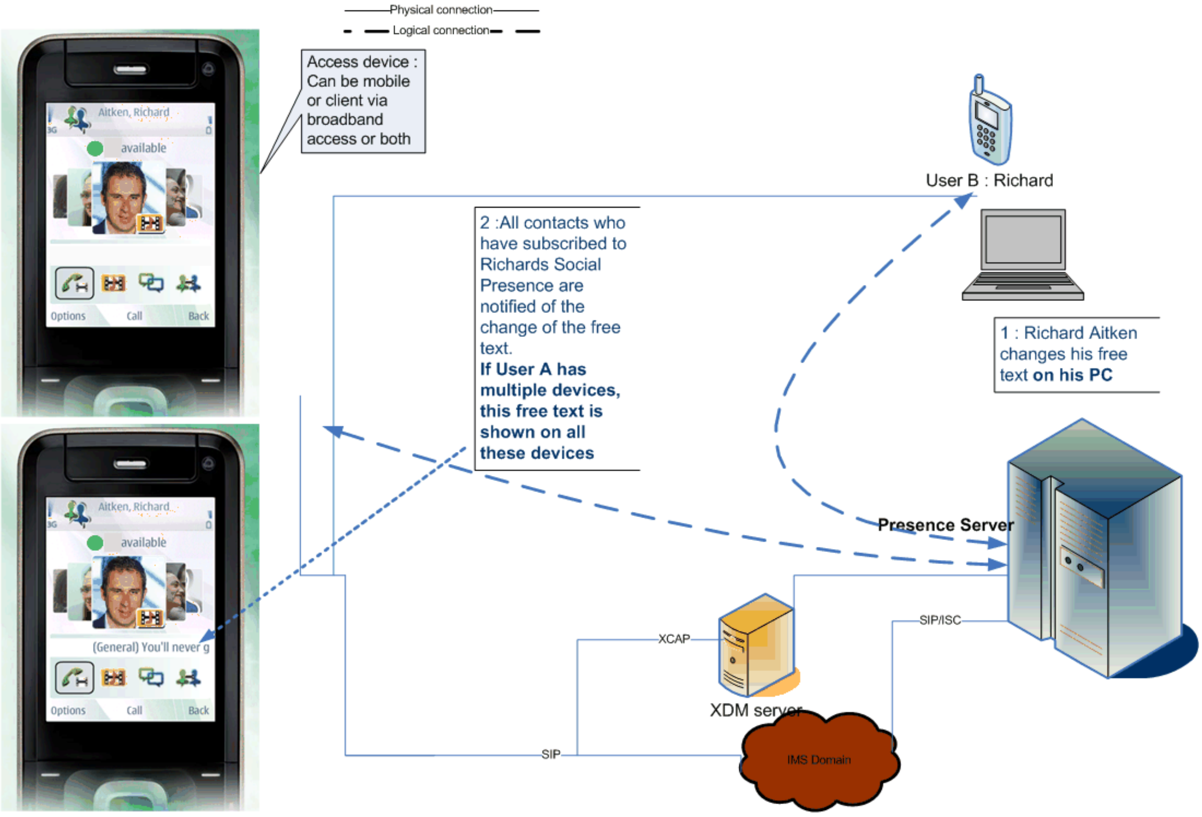

Figure 84: Free Text

NOTE: If the contact that is updated is not part of the VIP group of the user, the updated SPI (Richard Aitken's in the use case) may not be seen immediately. They will only see it when either their client polls for updates of the non-VIP contacts or when they request for an update of the non-VIP contacts themselves. The user may even miss the update altogether if there is another update before the status of the non-VIP contacts is retrieved.

3.7.1.5.1.5 Portrait Icon Exchange

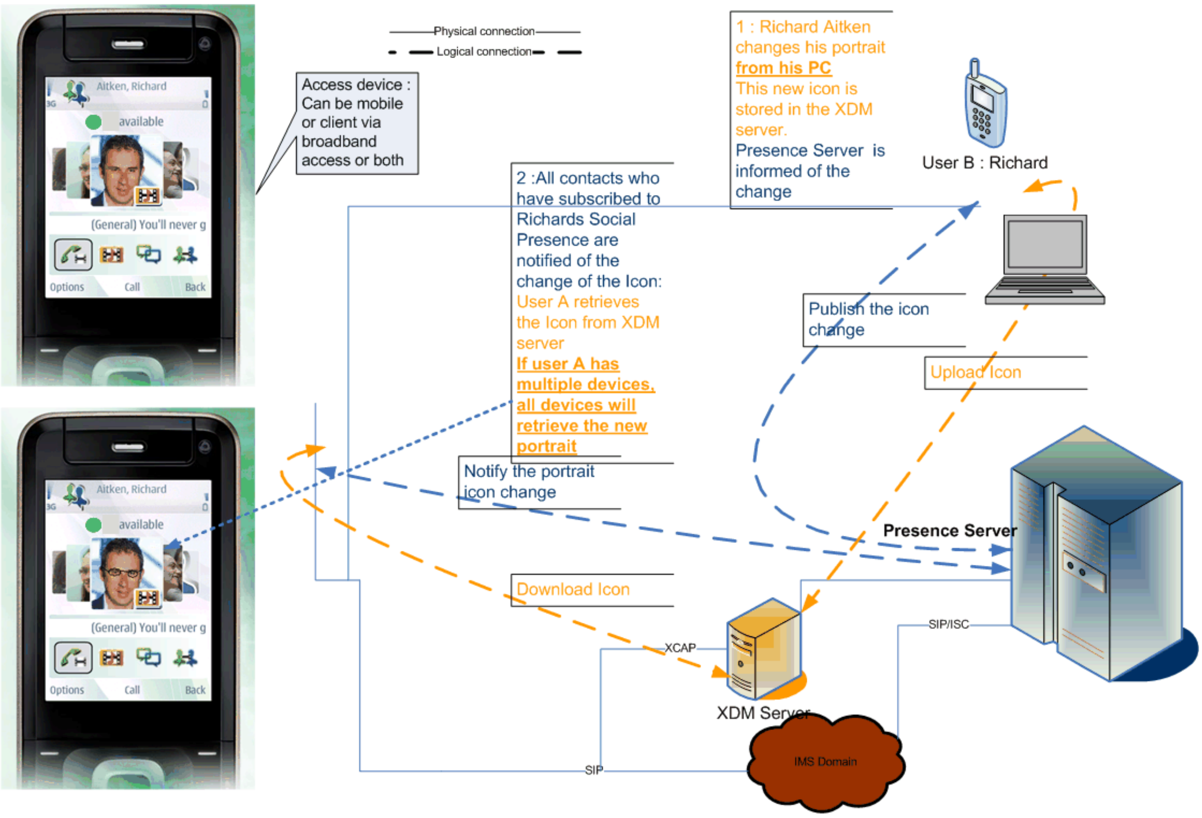

Figure 85: Portrait Icon Exchange

NOTE: If the contact that is updated is not part of the VIP group of the user the updated SPI (Richard Aitken's in the use case) may not be seen immediately. They will only see it when either their client polls for updates of the non-VIP contacts or when they request for an update of the non-VIP contacts themselves. The user may even miss the update altogether if there is another update before the status of the non-VIP contacts is retrieved.

3.7.1.5.1.6 Who Can I Invite?

New user wants to invite their friends to share social presence.

- User A goes to their RCS enhanced address book.
- User A traverses through the list of contacts and sees that User B is also an RCS user that supports the SPI service based on the capability discovery mechanism defined in section 2.6.
- User A decides to send an invitation to share Social Presence Information to User B.

3.7.1.5.2 Personalised Invitation with a Nickname

3.7.1.5.2.1 User A invites User B and fills out their Nickname. User A is present in User B's address book

- When User B receives the invitation, it is the contact name entered in User A's v-card that is used, not the nickname.

- For example, User B can read “<User A v-card name> <MSISDN> wants to share presence information with you.”

3.7.1.5.2.2 User A invites User B and fills out their Nickname. User B has not created a contact card for User A in their address book

- When User B receives the invitation, the nickname is used to present the invitation to User B
    - For example, User B can read “<User A nickname> <MSISDN> wants to share presence information with you.”
- If User B accepts the invitation, a contact card is created. User A’s nickname can be used to reference the contact card in User B’s address book.

3.7.1.5.3 Geolocation

3.7.1.5.3.1 Manual Free Text

- User A set his location manually (for example, I am in Paris).
- User B sees that User A is in Paris.

3.7.1.5.3.2 Manual Position on a Map

- User A decides to update their current location. User A drags and drops a pin on a map and then confirms the position. Eventhough User A is located in Paris, France, they select New York as a location on the map.
- User B receives a notification.
- User B sees that User A is in New York.

3.7.1.5.3.3 Semi-Automatic Filling

User A decides to edit their current location status. User A selects the location update button, and their location is automatically filled in the dedicated field used to enter their location.

3.7.1.5.3.4 Fully Automatic Opt-In Mode

User A decides that they want their authorised contacts to be informed regarding their position on a regular basis (period to be defined), they click on the “authorise my contacts to view my location” button (opt in). If they decide to end this broadcast they always have the ability to opt out through the same button.

In all cases, User B (authorised contact in User A’s address book) is notified as he would be notified of other presence information, such as status text.

3.7.1.5.3.5 Blocking an Authorised Contact from Viewing Location

- User A and B are authorised RCS contacts who have updated their location information.
- User A decides to hide their location from User B, while still sharing it with his other authorised contacts.

- User A goes to his location settings currently set to “Share my location with all my authorised contacts” to “Prevent some authorised contacts from viewing my location”.
- User A adds User B in the list of contacts blocked from viewing their location.
- User B does not see User A’s location information anymore.
- User A still sees User B’s location.

3.7.1.5.4 VIP Contacts

3.7.1.5.4.1 User sets a contact as a VIP

User A is an RCS user.

User B is an RCS user.

User A and User B had established a Social Presence Information sharing relationship.

Call Flow:

- User A sets User B Contact as a VIP contact in their address book.
- User B changes their Social Presence Information.
- User A receives an active notification (phone buzz or light, idle screen widget) about the change.

3.7.1.5.4.2 User sets a contact as a non-VIP

User A is an RCS user.

User B is an RCS user.

User A and User B had established a Social Presence Information sharing relationship.

User A had previously set User B as a VIP contact.

Call Flow:

- User A sets User B Contact as a non-VIP contact in their address book.
- User B changes their Social Presence Information.
- User A does not receive any active notification about the change but if they access later their EAB and browse to the User B contact, the EAB will display the changed information.

3.7.2 Interaction with other RCS features

Social Presence information in the device is linked with the local address book available in the device:

The social information elements of a contact in an RCS device are, from user interface point of view, associated (as an extension of other address book contact information) with the contact entry of the address book.

This correlation is local:

- Local contact information may be synchronised with a Network address book

- Extended presence information is obtained through the Network Presence enabler.

3.7.3 High Level Requirements

3-7-1 An RCS user with broadband access shall be able to access the Enhanced Address Book, supporting all the social presence features.
3-7-2 A broadband access client should support Social Presence Authorisation.
3-7-3 The presentity shall be able to edit the Social Presence Information from any of the devices he/she has and shall see the changes from every device he/she has.
3-7-4 Social Presence Information shall be handled in such a way that the latest update is presented to the watching user's client.
3-7-5 The invitation to share Social Presence Information shall be shown in all of the presentity's devices.
3-7-6 The presentity shall be able to authorise watchers from any of the devices they have
3-7-7 If a certain setting may limit the user experience provided to the end user, this information should be clearly shown in the user interface. In addition this allows the user to be aware of this limit while interacting with the service (for example, maximum number of characters to be included in the free text of the Social Presence Information, or maximum size of a file to be transferred).
3-7-8 The User shall be able to share location information as social presence information with his/her authorised contacts.
3-7-9 The User shall be able to define a list of contacts blocked from viewing his/her location information, within his list of authorised contacts for presence.
3-7-10 The User shall be able to specify their location through manual or automatic modes, as free text or as coordinates on a map.
3-7-11 The User shall be able to de-activate automatic updates or delete their location information at any time, to protect their privacy.
3-7-12 The User shall be able to share location information even if he/she is using a non-GPS device
3-7-13 The Service Provider shall be able to limit the frequency of automatic updates to avoid network overload.
3-7-14 The RCS user shall be able to set an expiration date for location information.
3-7-15 The User shall be able to define a nickname transmitted to his contacts when sending invitations, in addition to the MSISDN.
3-7-16 The User shall be able to change that nickname at any time, especially before sending invitations.
3-7-17 The Service Provide shall be able to specify the maximum length of the nickname.
3-7-18 The Nickname shall never automatically replace the existing registered name of a contact in the invitation recipient's phonebook.
3-7-19 The User shall be able to specify a text label displayed in lieu of the personal URL.
3-7-20 The User shall be able to change the URL label at any time.
3-7-21 Void.
3-7-22 An RCS user shall be able to set a contact as a VIP contact.
3-7-23 An RCS user shall be able to unset a contact as VIP.
3-7-24 When a VIP contact updates his Social Presence Information the user shall get a real time notification of the change and it shall be displayed on the RCS client (phone buzz or light indication, idle screen widget).
3-7-25 When a non-VIP contact updates their Social Presence Information, the user shall not be notified in real time about the changed status. The RCS client shall keep that information up to date (but not in real time) so the contact information is updated when the user browses the EAB.
3-7-26 The update mechanism for updating non VIP contacts shall be a periodic polling mechanism from the RCS client resulting in an aggregated notification from the

network. The update period shall be configured by the RCS Service Provider by parameter.

3-7-27 In addition, an RCS user shall be able to manually request an update of all the non-VIP contacts.

3.7.4 Technical Realisation

3.7.4.1 Network architecture of Presence enabler in RCS

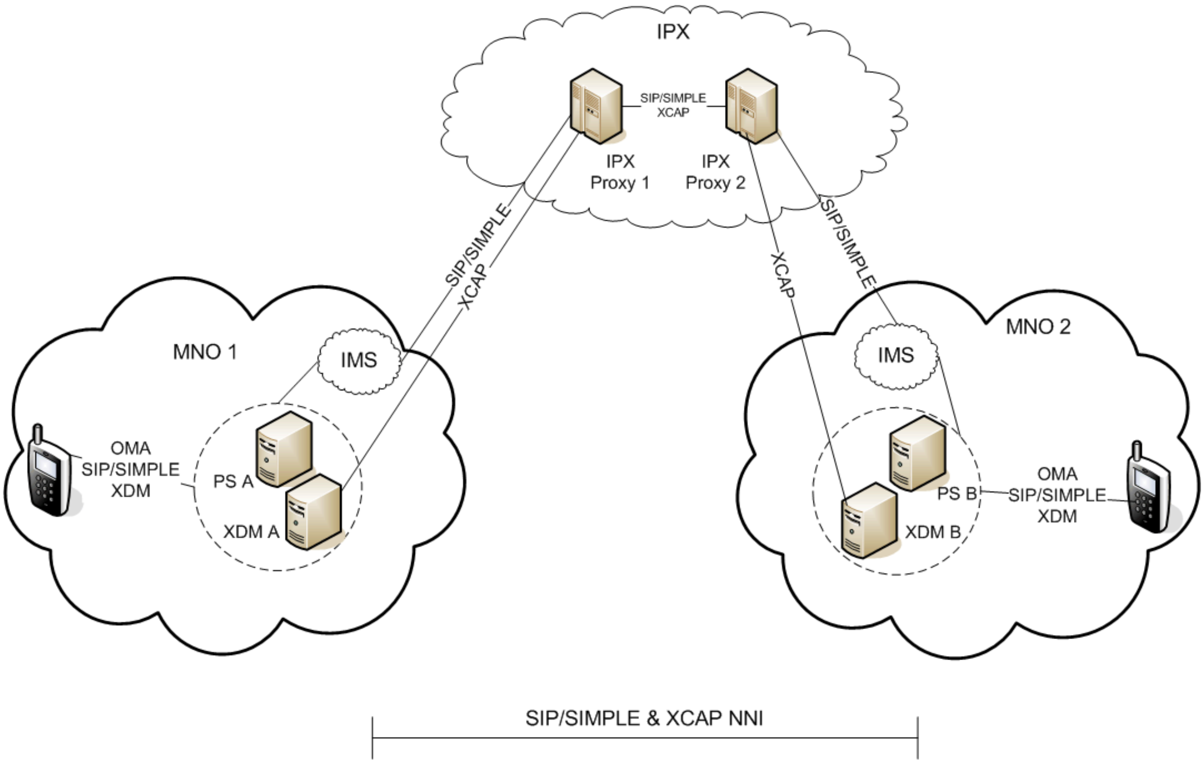

Figure 86: Overall Architecture of Presence as a part of RCS

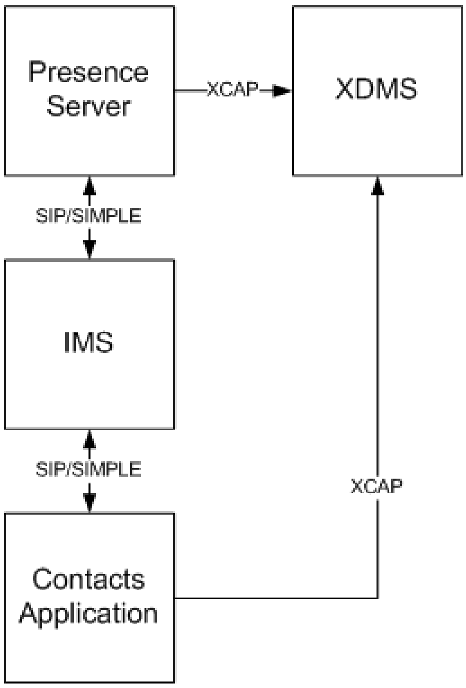

Figure 87: RCS Presence Architecture

Presence and capability architecture in RCS is based on [Presence].

Users share their Social Presence Information ("Presence Enhanced Address Book").

- Implemented using the Presence SIMPLE protocol

Users share their communication capability information ("Capability Enhanced Address Book").

- Can be implemented using the Presence SIMPLE protocol (see section 2.6.1.2).

According to [PRD-IR.65], the interworking connection should be carried out via IMS core systems. There is, therefore, no requirement to interface Presence Servers directly.

Optimisation of Presence & XDM enabler according to work in OMA PAG working group has to be taken into account as a very important design principle. It is also important to notice potential issues such as battery drain in the terminal caused by the general always-on functionality and the number of Presence & capability updates.

Generally, the Shared XDMS (XDM server) as defined in [XDM1.1_AD] shall be used for storing all presence-related lists, for example, the list of subscribed contacts ("buddy" list) and the presence authorisation lists. In this way, the RCS client only needs to operate on lists in Shared XDMS, and initially set the documents in RLS (Resource List Server) XDMS and Presence XDMS.

3.7.4.2 Presence Data Model

3.7.4.2.1 Overview

Implementation guidelines for the size/length of Presence information elements given in [PRESENCEIG] should be followed.

The following sections illustrate the details of the *Person* and *Device* parts of the Presence Data Model. The Service part of the model has been described in section 2.6.1.2.5.

3.7.4.2.2 Person

Attribute	Specification	Comment
Person: <presence> -> <person>	[Presence2.0_DDS]	According to the presence schema defined in the [Presence], person related information is modelled with the person element. Each client only publishes one person element.
Willingness: <person> -> <overriding-willingness> -> <basic>	[Presence2.0_DDS]	The presentity terminal publishes this attribute in which it wants to indicate its willingness to communicate: "Open" = Willing "Closed" = Not Willing Attribute not present = Unknown

Attribute	Specification	Comment
Icon: <person> -> <status-icon>	[Presence2.0_DDS]	It is used as dynamic avatar. If the element is not present the client may choose to display icon stored in the address book. The picture shall not be included directly in the presence requests, but a HTTP URL shall be used. Presence Content XDMS procedures as specified in OMA Presence 2.0 and XDM 2.0 is used for uploading, publishing and retrieving the icon For further details, see section 3.7.4.2.2.2
Favourite Link : <person> -> <link>	[Presence2.1_DDS]	The <link> element provides a URI pointing to general information about the tuple or person, typically a web home page. This information is complemented with a “label” attribute set to a value provided by the served RCS presentity and a priority attribute which is intended to cope with situations in which there are multiple <link> elements. In RCS, only one such <link> element will be included in the presence document though. The priority attribute will therefore always be set to 0.8.
Descriptive Location Text <person> -> <place-type> -> <other>	[Presence2.0_DDS]	The presentity may provide a descriptive text describing his location See section 3.7.4.2.2.3 for more information on the handling of the expiry of this information NOTE: Support for the enumerated values defined in [RFC4589] is thus out-of-scope for RCS. It is out of scope of RCS how a client will handle these enumerated values when received nevertheless.
Time Zone <person> -> <time-offset>	[Presence2.0_DDS]	The presentity may use this element to provide information on his current time zone See section 3.7.4.2.2.3 for more information on the handling of the expiry of this information
Geographical Information <person> -> <geopriv> -> <location-info> -> <usage-rules>	[Presence2.0_DDS]	This element can be used to provide geographical location information on the presentity. The accuracy of which can be controlled by the user. See section 3.7.4.2.2.3 for more details on its encoding and on the handling of the expiry of this information.
Note: <person> -> <note>	[RFC4479]	The presentity may write a piece of free text and/or to add emoticons to be shown to watchers in their contacts books. The list of emoticons in RCS can be found in [RCS-SIMPLEIM-ENDORS].
Timestamp: <person> -> <timestamp>	[RFC4479]	Timestamp when the presence information was published.

Table 45: Presence data model attributes

NOTE1: “Willingness” is sometimes indicated in a client as “Availability”. However, since it is managed by the user themselves and does not imply that communication is not possible within OMA specifications, this is considered as willingness. Availability indicates that on a technical level communication will be possible. Service Availability and Willingness are study items for later releases.

NOTE2: The priority of 0.8 for the link was included to allow including links with higher priority in some future RCS release.

3.7.4.2.2.1 Willingness

The RCS client will include in the presence document an OMA *<overriding-willingness>* element as specified in [Presence2.0_DDS] with the *<basic>* sub-element set to “closed” when the user has indicated that he’s not willing to communicate. Otherwise, if willingness is enabled, the published presence document will indicate a value of “open” for the *<basic>* sub-element of *<overriding-willingness>*.

3.7.4.2.2.2 Icon

The icon shall have following characteristics:

Document Name	rcs_status_icon
Icon aspect ratio (width:height)	3:4 or 4:3
Icon maximum dimensions	240x320
Icon minimum dimensions	60x80
Icon file type	gif (Graphics Interchange Format, both static and animated), jpeg (Joint Photographic Experts Group) or png (Portable Network Graphics) as defined in [Presence_Content]
Document maximum size	200 kilobytes (KB)

Table 46: Characteristics of the icon

NOTE1: Fixing the icon document name will ensure that for RCS usage, a single icon is stored in the network and no unnecessary resources are required for the storage of multiple icons. Without this, the situation could occur that multiple icons are stored without possibility to manage them after a switch to a new client. Furthermore, the fixing of the icon name will allow clients that are aware of the SIP URI of their contact to build the URI needed for the retrieval of the icon even if the contact is offline.

NOTE2: 200KB is not a mandatory size. It is only defined as a maximum and smaller sizes are acceptable

The other parameters are fixed to allow the client implementations to know what to expect.

3.7.4.2.2.3 Location Information

RCS clients shall not include a “from” attribute in the <place-type> and <time-offset> elements. RCS clients shall ignore it when received. RCS clients shall provide an "until" attribute in those elements and set it as specified in section 3.7.4.3.2.4.2.

RCS clients shall not include the optional description attribute in the <time-offset> element as this overlaps with the Location Type. RCS clients shall ignore it when received.

The geographical information will be provided as geographic coordinates. As specified for the “Geographical Location” building block in [Presence2.0_DDS], encoding will use the *<geopriv>*→*<location-info>* and *<geopriv>*→*<usage-rules>* elements.

The mandatory *<usage-rules>* element shall contain only a "*retention-expiry*" element as RCS clients will request the watchers to follow the default handling for the other rules. The RCS client shall set the "retention-expiry" as specified in section 3.7.4.3.2.4.2.

The *<location-info>* published by an RCS presence source will contain geographical information using the GML (Geography Markup Language) 3.1.1 Feature Schema (see [GML3.1.1]) which is the mandatory format to be used in the *<location-info>* element. The civic location format shall not be used by RCS presence sources and location information encoded in that way will be ignored by RCS clients when received.

RCS presence sources will within the *<location-info>* element represent an exact position by providing a GML *<point>* element and an inaccurate position as a *<circle>* element, both referring to the EPSG::4326 spatial reference schema as described in [RFC5491]. The coordinates of either the centre of this circle or the exact position will be represented with a single GML *<pos>* element with the actual coordinates as value. The radius of the circle will be represented in meters, which will be indicated by setting the unit of measure attribute of the radius element to the value of EPSG::9001 as described in [RFC5491]. An RCS client shall ignore any other type of data provided in the *<location-info>* element.

The European Petroleum Survey Group (EPSG) format requires that the coordinate representation is defined by the coordinate supplier. RCS presence sources will always provide the coordinates in WGS 84 (latitude, longitude) decimal notion as described in [RFC5491], providing the latitude and longitude as “double”-encoded decimal numbers (as specified in [GML3.1.1]) representing the degrees, separated by a space starting with the latitude. Negative values represent Southern and Western hemisphere respectively.

3.7.4.2.3 Service

See section 2.6.1.2.5.

3.7.4.2.4 Device

The Device part of presence is out of scope for RCS.

3.7.4.2.5 Example Document

The above leads to following example document:

```
<?xml version=”1.0” encoding=”UTF-8”?>
<presence xmlns=”urn:ietf:params:xml:ns:pidf”
          xmlns:op=”urn:oma:xml:prs:pidf:oma-pres”
          xmlns:opd=”urn:oma:xml:pde:pidf:ext”
          xmlns:opd11=”urn:oma:xml:pde:pidf:ext:1.1”
          xmlns:pdm=”urn:ietf:params:xml:ns:pidf:data-model”
          xmlns:rpid=”urn:ietf:params:xml:ns:pidf:rpid”
          xmlns:gp=”urn:ietf:params:xml:ns:pidf:geopriv10”
          xmlns:caps="urn:ietf:params:xml:ns:pidf:caps"
```

```
        xmlns:gml="http://www.opengis.net/gml"xmlns:gs="http://www.opengis.net/pidflo/1.0"
        entity="tel:+1234578901">
<tuple id="a2">
    <status><basic>open</basic></status>
    <op:service-description>
        <op:service-id>org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel</op:service-id>
        <op:version>1.0</op:version>
    </op:service-description>
    <caps:servcaps>
        <caps:audio>true</caps:audio>
        <caps:duplex>
            <caps:supported>
                <caps:full/>
            </caps:supported>
        </caps:duplex>
    </caps:servcaps>
    <contact>tel:+1234578901</contact>
</tuple>
<tuple id="a1">
    <status><basic>open</basic></status>
    <op:service-description>
        <op:service-id>org.3gpp.cs-videotelephony</op:service-id>
        <op:version>1.0</op:version>
    </op:service-description>
    <contact>tel:+1234578901</contact>
</tuple>
<tuple id="a12">
    <status><basic>open</basic></status>
    <op:service-description>
        <op:service-id>org.gsma.videoshare</op:service-id>
        <op:version>1.0</op:version>
    </op :service-description>
    <contact>tel:+1234578901</contact>
</tuple>
<tuple id="a123">
    <status><basic>open</basic></status>
    <op:service-description>
        <op:service-id>org.gsma.videoshare</op:service-id>
        <op:version>2.0</op:version>
    </op :service-description>
    <contact>tel:+1234578901</contact>
</tuple>
<tuple id="a132">
    <status><basic>open</basic></status>
    <op:service-description>
        <op:service-id>org.openmobilealliance:IM-Session</op:service-id>
        <op:version>1.0</op:version>
    </op :service-description>
    <contact>tel:+1234578901</contact>
</tuple>
<pdm:person id="a1233">
    <op:overriding-willingness>
        <op:basic>open</op:basic>
    </op:overriding-willingness>
    <rpid:status-icon opd:etag="26362">http://xcap.gsma.org/xcap-ap/service/org.openmobilealliance.pres-
    content/users/sip:1234578901@gsma.org/oma_status-icon/rcs_status_icon</rpid:status-icon>
    <opd11:link opd11:label="my blog" opd11:priority="0.8">
        http://example.com/~alice
```

```
      </opd11:link>
      <rpid:place-type opd:until=”2009-11-28T21:00:00Z”>
          <rpid:other>Herentals, Belgium</rpid:other>
      </rpid:place-type>
      <rpid:time-offset opd:until=”2009-11-28T21:00:00Z”>+120</rpid:time-offset>
      <gp:geopriv>
          <gp:location-info>
              <gs:Circle srsName="urn:ogc:def:crs:EPSG::4326">
                <gml:pos>51.1644 4.7880</gml:pos>
                <gs:radius uom="urn:ogc:def:uom:EPSG::9001">10</gs:radius>
              </gs:Circle>
          </gp:location-info>
          <gp:usage-rules>
              <gp:retention-expiry>2009-11-28T21:00:00Z</gp:retention-expiry>
          </gp:usage-rules>
      </gp:geopriv>
      <pdm:note>I’ll be PAG</pdm:note>
  </pdm:person>
</presence>
```

Table 47: Example Presence Document

3.7.4.3 Presentity Side Handling

3.7.4.3.1 Publication Methods

3.7.4.3.1.1 Overview

An RCS client publishes its presence information using two different methods:

1. SIP PUBLISH requests.
2. Permanent Presence State Publication (that is, a permanent document maintained through XCAP).

The method to be used depends on the information to be published:

SIP PUBLISH requests are used for following data:

- Service Capabilities.

Permanent Presence State publication applies to the following attributes of Social Presence Information:

- Portrait icon.
- Free text.
- Favourite link.
- Willingness (that is the overriding-willingness element).
- Location Information.

3.7.4.3.1.2 Permanent Presence State Publication

The RCS Client shall support Permanent Presence State publication by manipulating the Permanent Presence State via an XDMC using the permanent presence state application as defined in [Presence2.0_TS]. An RCS client shall update the permanent presence state document in such a way that elements in the document that are not changed or are even unknown to the RCS client (for example, because they were included by a client supporting a future RCS release), are not altered. To avoid inconsistencies between attributes and the actual element value, unknown attributes of changed elements shall be removed from the updated document.

This can be achieved both through a direct, conditional update of only the changed element itself or through a retrieval of the complete document followed by a client local update of the changed elements. This update should then be used in a conditional replace request for the entire permanent presence state document. The choice between both methods is left to client implementation and could depend on the amount of updated elements. In both cases, whenever the document is modified any expired information will be removed (for example Location Information with an “until” attribute indicating a time in the past).

The RCS Presence Server shall use the Permanent Presence State as input for Presence Information processing. RCS Presence Server should subscribe/fetch the permanent presence state document from Presence XDM when applying the composition policy.

3.7.4.3.2 Presence Information Handling

3.7.4.3.2.1 Willingness

At the presentity side, the RCS client will always include an <*overriding-willingness*> element in the permanent presence document. This element will have a <*basic*> sub-element set to either “*open*” or “*closed*” depending on what was indicated by the user as his current status. If willingness is disabled through the provisioning parameter, no “<*overriding-willingness*>” element will be included in the permanent presence document.

3.7.4.3.2.2 Status Icon

The status icon shall be stored, updated, deleted and retrieved according to the OMA Presence and XDM 2.0 procedures. For the storage itself, the Presence Content XDMS as defined in [Presence_Content] shall be used including the application usage and document type that it introduces. RCS will only make use of the Presence Content XDMS for the storage of the status icon. Therefore the usage as defined in section 5.1.12.1 of [Presence_Content] is the only one that is applicable including all its associated restrictions. After storing, updating or deleting the icon, the presentity’s client should publish an updated presence document including the *etag* attribute in the <*status-icon*> element as described in [Presence2.0_DDS] in sections 7.11.1.3 and 7.20.

3.7.4.3.2.3 Link

The RCS client will limit the length of the label 200 characters.

3.7.4.3.2.4 Location

3.7.4.3.2.4.1 Ending Location Information Sharing

When the user indicates that they do not want to share their location information with the contacts allowed to see their information anymore, the client can fulfil this request by removing the location information from the Permanent Presence State document.

3.7.4.3.2.4.2 Managing Location Information

An RCS presence source is not required to include all location elements specified in section 3.7.4.2.2.3 in the permanent presence state document (that is, all elements are optional to be provided).

The length of the descriptive text that the RCS client includes in the Permanent Presence State document shall not be longer than 200 characters.

The maximum time a location update remains available to watchers is controlled by a Service Provider provisioning setting. RCS presence sources will set the "*until*" attribute and the "*retention-expiry*" element (see section 3.7.4.2.2.3) in accordance to this provisioning setting (that is, set it to the current time increased with the value of the setting). Furthermore, RCS presence sources shall remove expired location information from the published presence document and from any locally cached copy of that document whenever they update other elements in the document.

Clients offering the user the choice to provide an inaccurate position to their contacts (for example, city level or even country level) can do so by providing a *CircleByCenterPoint* element instead of an exact position using coordinates and text reflecting this inaccuracy (for example, the city centre instead of the exact street). Whether the client does this and how it determines the position of the centre, the radius and the text value (that is, the *<place-type>* element) that will be shared, is considered to be client implementation and thus out-of-scope for RCS.As an option to the user, clients may also offer the possibility to regularly update their position without user intervention. Whether or not this is done is again considered to be a client implementation issue and thus out-of-scope for RCS.

3.7.4.3.2.5 Nickname

The *application/watcherinfo+xml* body in the watcher information notification may contain a display name for the watcher in the display-name attribute as specified in [RFC3858]. In this case, if the telephone number that is derived from the (SIP or tel) URI that is provided for that watcher is not found in the phone book of the client, the RCS client will include the display name in notifications shown to the user. At the same time, it will always include the watcher's telephone number to minimise the risk of false identifications.

If no display name is received (for example, because the subscription is initiated from an RCS Release 2 network), the client shall only present the E.164 number to the user.

If the watcher's telephone number is found in the phone book, behaviour shall be as specified in section 2.5 (that is, the received display name shall not be used, but rather the information that is part of the phone book).

An RCS client shall be able to deal with display names up until a maximum length of 200 characters.

3.7.4.3.3 Multidevice Handling

If one of the user's clients changes the (shared) permanent presence state document, the other clients of the user will receive the update as part of a presence notification which will contain information about their own presentity. Such an update will be received immediately when the client is online at the time of the changes. If this is not the case, the client will receive the update when it comes online. Clients shall take the updated social presence information into account and update the presence information that they store locally in the client accordingly. To get the notifications that are necessary to provide this behaviour, the client shall include the own identity in the "*rcs*" list which is part of the Shared XDMS's "*resource-lists*" document (see section 2.14.1).

When a user decides that they do not want to receive a certain service on one of their secondary devices (see section 2.9.1.4), the client on the given secondary device will not indicate the capability for that service in the services section of the presence document if such a capability is defined for the service (see section 2.6.1.2.5).

3.7.4.4 Watcher Side Handling

When presence information of a presentity is requested by a watcher a SUBSCRIBE request is initiated (event package '*presence*') according to [Presence]. The watcher should be able to use the tel URI to identify the presentity, see section 2.5.

The support of RLS is mandatory for the clients and servers. Client shall conform to section 5.2.2.1 of the technical specification of [PRESENCE] and in addition to section 5.7.1 and 5.8 in [PRESENCEIG], section 5.1 in [XDMIG] and section 5.1.6 in [RLSXDM]. The XML documents shall follow the templates following later in this section.

3.7.4.4.1 Caching Presence Information

The caching of presence information is a client procedure.

The RCS client must be able to locally store the most up-to-date presence information (that have been received through notifications) of all of the user's contacts. This locally stored information must be handled as a persistent cache (that is the data shall not be erased when the terminal is switched-off).

3.7.4.4.2 Presence Information Handling

3.7.4.4.2.1 General Processing Rules to Facilitate Forwards Compatibility

To maintain enough flexibility and not to impose potentially sub-optimal technical choices on future RCS releases, the presence parsing for social presence information in an RCS client should be sufficiently robust. Therefore the following guidelines should be taken into account in RCS presence parsing:

- Unknown or unsupported elements could be present in the document. In that case they should be ignored.
- When using RLS subscriptions, information could be contained on presentities that were not known to be part of the presence list (for example because the list was updated by another client or application). If the unexpected presentity is a known contact, the client should treat this contact as being presence enabled (see section 3.7.4.4.4) and try to retrieve an updated presence list from the network (see section 2.14.2).
- The Watcher shall follow the procedures defined in section 6.2 "Default Watcher Processing" of [Presence2.0_DDS].

3.7.4.4.2.2 Willingness

The client will interpret any "<*overriding-willingness*>" element included/not included in the received presence document(s) as specified in section 3.7.4.2.2.

3.7.4.4.2.3 Status Icon

The link to the status icon that is received in the presence document of the contact will be processed as described in [Presence2.0_TS] section 5.2.5.3. When the *etag* attribute of the *status-icon* element does not match that of the cached icon, the client will download the updated icon. To do that it will handle the link that it received in the presence document as defined in [XDM2.0_Core] section 6.1.1.1 and more specifically the third paragraph: it will replace the XCAP root part of the link with the own XCAP root of the watcher. After downloading the icon, the RCS client shall cache it along with the *etag* to be able to process future notifies on the status of the contact as defined in [Presence2.0_TS] section 5.2.5.3.

3.7.4.4.2.4 Link

If an RCS client receives a document containing multiple <*link*> elements, then it shall only consider the one with the highest priority and use that as the value of the <*link*> element in the processing.

An RCS watcher shall be able to deal with labels with a length of maximum 200 characters.

3.7.4.4.2.5 Location Information

It is considered to be a client implementation decision how received location information from a contact will be handled (for example, display only the text, use an individual map for each contact and so on. This is thus considered to be out of scope for RCS. Clients should at least provide a means to display any descriptive text (that is, the content of the <*place-type*> element) that they might receive.

An RCS client should take into account that a received presence document might not contain location information (for example, because the presence source does not provide it or privacy was enabled).

An RCS client shall be able to deal with *place-type* information with a length of maximum 200 characters.

An RCS client shall not display to the user information contained in location elements for which the "*until*" attribute (for the <*time-offset*> and <*place-type*> elements) or the <*retention-expiry*> element (for the geolocation information) indicate a time in the past. Furthermore it shall not cache the expired information locally any longer.

3.7.4.4.3 Nickname Handling

If the user has provided a nickname, an RCS client shall include it as the display name as part of the identity information provided in the *P-Preferred-Identity* and *From* header field of the SIP SUBSCRIBE request used when subscribing to the user's Resource List Server (RLS) document. The RCS client shall ensure that the length of the used display name is not larger than the maximum size that was provisioned by the Service Provider.

3.7.4.4.4 Multidevice Handling

For the most part the watcher functionality on the different clients of the same user can function independently of each other. Only with the authorisation there might be some interaction as this may trigger unexpected notifications (see section 3.7.4.5.7). An RCS client of this release will provide compatibility with clients of future RCS releases acting as one or more of the other devices of the user. To achieve this it will display the presence information provided in a presence notification if it refers to a known contact, regardless of whether that contact can be found in the "*rcs*", "*rcs_basic_spi_only*", "*rcs_poll*" or "*rcs_poll_basic_spi_only*" lists of the Shared XDMS's "*resource-lists*" document (see section 3.7.4.5.2).

3.7.4.5 Subscriptions and Authorization

3.7.4.5.1 Overview

Presence invitations are subject to reactive authorisation to guarantee user privacy. This will allow the invited user (presentity) to accept, block or ignore an invitation to establish a presence relationship.

The presence authorisation for basic social presence information shall be symmetric. This means the inviting user automatically authorises the invited user to see their basic social presence information. The invited user by accepting the presence invitation request both authorises the inviting user to see their basic social presence information and subscribes to the inviting users presence information.

The RCS presentity shall be able to configure the presence authorisation rules, which require the support in the RCS client and in the RCS Presence Server of [PresenceXDM]. The RCS client shall store a presence authorisation document that follows [PresenceXDM] and the template rules described in section 5.8 in [PRESENCEIG].

In order for a presentity to be able to authorise the subscription of a watcher, the presentity needs to know which watcher(s) are trying to subscribe to the presence of the presentity. The RCS client and the Presence Server shall thus support section 5.3.1 and 5.4.4 of [Presence].

When the subscription is authorised successfully, the Presence Server sends the presentity's presence document to the watcher by using the NOTIFY method as defined in [Presence]. The format of the presence notification follows the Presence Data Model as describe above and it contains the information the watcher is allowed to see according to the configured presence rules.

The contacts with whom the RCS user share presence information can be defined as either VIP contacts or non-VIP contacts (see section 3.7.1.4.9). For VIP contacts, presence information changes are received in real time, using a subscription to the corresponding "VIP contacts" buddy list in RLS. For non-VIP contacts the client will poll the corresponding "non-VIP contacts" list in RLS to retrieve presence information changes.

Contrary to the general concept for basic social presence information sharing the authorisation for location information is not necessarily mutual: User A can get the location information from User B without having to provide his location information. Furthermore, the user can control whether the information that he/she is capable of sharing social presence information is public or not.

3.7.4.5.2 XML Document Structure

The Presence XDMS shall contain the following authorisation rules following, where possible, the recommendations in [PRESENCEIG]:

- "*allow own*" rule – allows subscriptions to own presence data

- ”*confirm unlisted*” rule – allows reactive authorisation for contacts not yet allowed or blocked
- ”*blocked contacts*” – contains those contacts that the user has blocked (points to ”*blocked contacts*” list in Shared XDMS)
- ”*granted contacts*” rule – will be used as the rule to provide all social presence information (that is, the Basic Social Presence Information and geolocation information)
- “*basic_spi_only_granted_contacts*” rule – will be used by the contacts with whom no location information is being shared.

The RLS XDMS shall for an RCS user contain two entries; one referencing the “*oma_buddylist*” list and one referencing the “*rcs_poll_buddylist*” list, both in Shared XDMS for which the template is described in section 2.14.1. The service URI referencing the “*oma_buddylist*” allows subscribing with one RLS subscription to the presence information of both the VIP contacts with whom only social presence information is shared and those VIP contacts that are also allowed to see the location information. The RCS client will at start-up subscribe to changes to this list by issuing a SUBSCRIBE request to the RLS targeting this list with an expire value >0 (pre-configured in client).

In addition to information on the VIP contacts, the service URI referencing the “*rcs_poll_buddylist*” (see section 2.14.1 for the template) allows the RCS client with one subscription request to retrieve presence information also from the non-VIP contacts with whom only social presence information is shared and those non-VIP Contacts that are also allowed to see the location information. The RCS client will, only on user request or also on regular basis issue a “poll” SUBSCRIBE (that is with expires=0) to this list to obtain the presence information for the contacts in this list.

The maximum amount of poll operations on the non-VIP Contacts buddy list during a certain time period can in the client be configured subject to Service Provider policies (see Annex A).

The Shared XDMS (see section 2.14.1 for the template) shall contain the following lists that are used for presence and are provided and managed by the RCS client:

- “*rcs*” list: This list includes all VIP contacts with which basic Social Presence and location information is shared. Commonly referred in RCS from both the “*oma_buddylist*” and “*oma_grantedcontacts*” lists as the contacts that are allowed to see your presence are also your buddies (symmetric).
  To provide the behaviour described in section 3.7.4.3.3, the “*rcs*” list will contain the own identity of the user. The client shall not allow the user to remove that entry.
- “*rcs_basic_spi_only*” list: This list includes all VIP contacts with which only basic Social Presence information is shared. Commonly referred in RCS from both the “*oma_buddylist*” and “*rcs_basic_spi_only_granted_contacts*” lists as the contacts that are allowed to see your presence are also your buddies (symmetric).
- “*rcs_poll*” list: This list includes all non-VIP contacts with which basic Social Presence and location information is shared. Commonly referred in RCS from both the “*rcs_poll_buddylist*” and “*oma_grantedcontacts*” lists as the contacts that are allowed to see your presence are also your buddies (symmetric). As a difference with the “*rcs*” list, the “*rcs_poll*” list will not contain the own identity of the user.

- "*rcs_poll_basic_spi_only*" list: This list includes all non-VIP contacts with which only basic Social Presence information is shared. Commonly referred in RCS from both the "*rcs_poll_buddylist*" and "*rcs_basic_spi_only_granted_contacts*" lists as the contacts that are allowed to see your presence are also your buddies (symmetric).
- "*oma_buddylist*" list: Contains a reference to the "*rcs*" and the "*rcs_basic_spi_only*" lists where the actual VIP Contacts (or buddies) are stored. The "*oma_buddylist*" is explicitly used from the RLS document.
- "*rcs_poll_buddylist*" list: Contains a reference to the "*rcs_poll*" and the "*rcs_poll_basic_spi_only*" lists where the actual non-VIP Contacts are stored. The "*rcs_poll_buddylist*" is explicitly used from the RLS document.
- "*oma_grantedcontacts*" list: This list includes all contacts you have authorised to see your basic social presence and location information. Contains a reference to the "*rcs*" and "*rcs_poll*" lists.
- "*rcs_basic_spi_only_grantedcontacts*" list: This list includes all contacts you have authorised to see only your basic social presence information. Contains a reference to the "*rcs_basic_spi_only*" and the "*rcs_poll_basic_spi_only*" lists
- "*oma_blockedcontacts*" list: Contains a reference to the "*rcs_blockedcontacts*" list where the actual permanently blocked contacts are stored and to the "*rcs_revokedcontacts*" list with the revoked users that are temporarily being blocked.
- "*rcs_blockedcontacts*" list: Contains all permanently blocked contacts
- "*rcs_revokedcontacts*" list: Contains all revoked contacts that are currently being blocked.

NOTE1: The "*rcs_revokedcontacts*" list is not intended to be shown to the end user. It is managed automatically.

NOTE2: A contact should only be in one of the lists used for presence. To ensure this, the RCS client shall check the other lists for an occurrence of the contact when adding it to a list. If the contact occurs somewhere else, the client will remove that entry. A contact will always be added to the new list before being removed from the old one. This applies both when removing a presence relation (see section 3.7.4.5.4) and when changing a contact from being a VIP Contact to a being a non-VIP Contact or vice versa (see section 3.7.4.5.6).

For RCS, the template definitions below will be used for the different XDM documents related to presence subscriptions and authorisations.

Presence XDMS:
AUID: org.openmobilealliance.pres-rules
Document name: pres-rules
Template

```
<?xml version="1.0" encoding="UTF-8"?>
<cr:ruleset
    xmlns:ocp="urn:oma:xml:xdm:common-policy"
    xmlns:op="urn:oma:xml:prs:pres-rules"
    xmlns:pr="urn:ietf:params:xml:ns:pres-rules"
    xmlns:cr="urn:ietf:params:xml:ns:common-policy">
    <cr:rule id="wp_prs_allow_own">
```

```
        <cr:conditions>
          <cr:identity>
              <cr:one id="tel:+1234578901"/>
          </cr:identity>
        </cr:conditions>
        <cr:actions>
          <pr:sub-handling>allow</pr:sub-handling>
        </cr:actions>
        <cr:transformations>
          <pr:provide-services>
              <pr:all-services/>
          </pr:provide-services>
          <pr:provide-persons>
              <pr:all-persons/>
          </pr:provide-persons>
          <pr:provide-devices>
              <pr:all-devices/>
          </pr:provide-devices>
          <pr:provide-all-attributes/>
        </cr:transformations>
    </cr:rule>

    <cr:rule id="wp_prs_unlisted">
        <cr:conditions>
          <ocp:other-identity/>
        </cr:conditions>
        <cr:actions>
          <pr:sub-handling>confirm</pr:sub-handling>
        </cr:actions>
    </cr:rule>
    <cr:rule id="wp_prs_grantedcontacts">
        <cr:conditions>
          <ocp:external-list>
              <ocp:entry anc="http://xcap.gsma.org/resource-
              lists/users/sip:1234578901@gsma.org/index/~~/resource-
              lists/list%5B@name=%22oma_grantedcontacts%22%5D"/>
          </ocp:external-list>
        </cr:conditions>
        <cr:actions>
          <pr:sub-handling>allow</pr:sub-handling>
        </cr:actions>
        <cr:transformations>
          <pr:provide-services>
              <pr:all-services/>
          </pr:provide-services>
          <pr:provide-persons>
              <pr:all-persons/>
          </pr:provide-persons>
          <pr:provide-devices>
              <pr:all-devices/>
          </pr:provide-devices>
          <pr:provide-all-attributes/>
        </cr:transformations>
    </cr:rule>

    <cr:rule id="rcs_basic_spi_only_grantedcontacts">
        <cr:conditions>
          <ocp:external-list>
```

```
                <ocp:entry anc="http://xcap.gsma.org/resource-
                lists/users/sip:1234578901@gsma.org/index/~~/resource-
                lists/list%5B@name=%22rcs_basic_spi_only_grantedcontacts%22%5D"/>
            </ocp:external-list>
          </cr:conditions>
          <cr:actions>
            <pr:sub-handling>allow</pr:sub-handling>
          </cr:actions>
          <cr:transformations>
            <pr:provide-services>
                <pr:all-services/>
            </pr:provide-services>
            <pr:provide-persons>
                <pr:all-persons/>
            </pr:provide-persons>
            <pr:provide-devices>
                <pr:all-devices/>
            </pr:provide-devices>
            <pr:provide-note>true</pr:provide-note>
            <pr:provide-status-icon>true</pr:provide-status-icon>
            <pr:provide-unknown-attribute
                ns="urn:oma:xml:pde:pidf:ext:1.1"
                name="link">
                true
            </pr:provide-unknown-attribute>
            <op:provide-willingness>true</op:provide-willingness>
            <pr:provide-unknown-attribute
                ns="urn:oma:xml:prs:pidf:oma-pres"
                name="service-description">
                true
            </pr:provide-unknown-attribute>
          </cr:transformations>
      </cr:rule>

      <cr:rule id="wp_prs_blockedcontacts">
          <cr:conditions>
            <ocp:external-list>
                <ocp:entry anc="http://xcap.gsma.org/resource-
                lists/users/sip:1234578901@gsma.org/index/~~/resource-
                lists/list%5B@name=%22oma_blockedcontacts%22%5D"/>
            </ocp:external-list>
          </cr:conditions>
          <cr:actions>
            <pr:sub-handling>block</pr:sub-handling>
          </cr:actions>
      </cr:rule>
</cr:ruleset>
```

Table 48: Presence Rules Template

NOTE: If the client is configured to use a presence based capability discovery (as described in section 2.6.1.2, the *rcs_allow_services_anonymous* rule described in Table 18 should be included in this template.

RLS XDMS:

AUID: rls-services

Document name: index

Template:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rls-services xmlns="urn:ietf:params:xml:ns:rls-services">
    <service uri="sip:1234578901@gsma.org;pres-list=rcs">
        <resource-list>http://xcap.gsma.com/services/resource-
        lists/users/sip:1234578901@gsma.org/index/~~/resource-
        lists/list%5B@name=%22oma_buddylist%22%5D</resource-list>
        <packages>
          <package>presence</package>
        </packages>
    </service>
    <service uri="sip:1234578901@gsma.org;pres-list=rcs_poll">
        <resource-list>http://xcap.gsma.com/services/resource-
        lists/users/sip:1234578901@gsma.org/index/~~/resource-
        lists/list%5B@name=%22rcs_poll_buddylist%22%5D</resource-list>
        <packages>
          <package>presence</package>
        </packages>
    </service>
</rls-services>
```


Table 49: Presence RLS Template

NOTE: The lists in the Shared XDMS and the general procedures on the handling of XDM requests and the creation of the documents are described in section 2.14.

3.7.4.5.3 Client Procedures, Initiation of Presence Sharing

When initiating a presence sharing request, the inviting user's RCS client adds the invited user's URI to the "*rcs*" list in Shared XDMS according to the procedures in [Shared-XDM].

When the invited user receives a notification to establish a presence relation, the user can either:

1. Accept the invitation, whereas the RCS client of the invited user adds the inviting User's URI to the "*rcs*" list in Shared XDMS (see section 2.14.1) according to the procedures in [SHARED-XDM].
2. Block the invitation, whereas the RCS client of the invited user adds the inviting User's URI to the "*rcs_blockedcontacts*" list in Shared XDMS (see section 2.14.1) according to the procedures in [SHARED-XDM].
3. Ignore the invitation, whereas the RCS client of the invited user removes the presence sharing invitation.
4. Not answer the invitation. The presence sharing invitation is pending in the client until either "accepted" (case 1), "blocked" (case 2) or "ignored" (case 3). In the signalling, there is no difference from the "ignore" case.

3.7.4.5.4 Client Procedures, Removal of Presence Sharing

When the user decides to end the presence relationship with one of their contacts, they have to use the revoke option on their device. This triggers a notification to the user as defined in section 3.7.1.4.6 asking for confirmation. When this is confirmed, the client will put the user on the "*rcs_revokedcontacts*" list, subsequently remove the user from the "*rcs*" or "*rcs_basic_spi_only*", "*rcs_poll*" or "*rcs_poll_basic_spi_only*" list and remove the contact's presence information from the cache as defined in section 3.7.4.4.1. When putting an entry for the contact in the "*rcs_revokedcontacts*" list the client includes a last modified attribute that indicates the current time in UTC.

When a client notices it has been blocked by a contact with whom Social Presence was shared (that is the RLS notify indicates the subscription is in state "*terminated*" and the reason indicates "*rejected*"), it will remove the contact from the "*rcs*" or "*rcs_basic_spi_only*", "*rcs_poll*" or "*rcs_poll_basic_spi_only*" list and remove the contact's cached presence information. Note that for a non-VIP contact (in the "*rcs_poll*" or "*rcs_poll_basic_spi_only*" list) there could be a delay in the detection of this change.

All clients will process the "*rcs_revokedcontacts*" list periodically and remove those contacts that have been included in the list for a sufficiently long period already (for example several days). For that they will compare the last-modified attribute of the entries to the current time. Both the interval at which the list is checked (REVOKE TIMER) and the period that a contact should remain in this list (REVOKE TIMER) is a Service Provider configurable client parameter defined in Annex A.

With regards to the communication capabilities both clients should fall back to the procedures as defined in section 2.6 for sharing of capabilities between contacts not sharing social presence information.

3.7.4.5.5 Conditional Event Notification

The support of conditional event notification is strongly recommended for the clients (i.e. Watcher and Watcher Information Subscriber) and for the servers (i.e. Presence Server and RLS) to optimize presence traffic at UNI and NNI.

An RCS client should support subscription with conditional event notification, as defined in section 5.2.6 and section 5.3.2 of [Presence2.0_TS].

An RCS RLS should support subscription with conditional event notification, as defined in section 5.2 of [Presence2.0_RLS_TS].

An RCS Presence Server should support notification with conditional event notification, as defined in section 5.5.3.8, 5.5.3.9 and 5.5.4.2 of [Presence2.0_TS].

An RCS RLS should support notification with conditional event notification, as defined in section 5.4 of [Presence2.0_RLS_TS].

3.7.4.5.6 Client Procedures, managing of VIP and non-VIP Contacts

When the user decides to change a user from being a VIP Contact to being a non-VIP Contact (or vice versa) the client will first add the user's URI to the target list and after this, remove the user's URI from the list where it was previously stored. That is, when changing a user from being a VIP Contact to a non-VIP Contact, the client will first add the user's URI to the "*rcs_poll*" (if previously in the "*rcs*" list) or "*rcs_poll_basic_spi_only*" list (if previously in the "*rcs_basic_spi_only*" list) and then remove the URI from the "*rcs*" or "*rcs_basic_spi_only*" list respectively. When changing a user from being a non-VIP Contact to a VIP Contact, the client will first add the user's URI to the "*rcs*" list (if previously in the "*rcs_poll*" list) or "*rcs_basic_spi_only*" list (if previously in the "*rcs_poll_basic_spi_only*" list) and then remove the URI from the "*rcs_poll*" or "*rcs_poll_basic_spi_only*" list respectively.

3.7.4.5.7 Multidevice Handling

Any negative effects of XDM document changes in a multidevice context are countered through the XDM document handling as it is described in section 2.14.2.

Several situations should be dealt with:

- The user owning multiple clients is invited by a contact to share social presence information.
  All the user's active clients will receive watcher information notifications both when the contact subscribes for the user's social presence information (subscription entering the "*pending*" state) and when the user accepts or blocks the "invitation" on one of their clients (subscription going out of the "*pending*" state). When the user accepts the invitation on one of their clients, the other clients will also start receiving the social presence information of the contact.
- The user owning multiple clients invites a contact to share social presence information from one of their clients.
  In this case their other clients will receive presence notifications indicating that a subscription to the contact entered the pending state and notifications including the other user's social presence information when the contact accepted the "invitation". If the contact blocks the "invitation", there will be presence notifications to all the user's clients indicating that the subscription was terminated. The clients shall use these unexpected notifications as triggers to update the locally stored copy of the Shared XDMS's "*resource-lists*" document if they cache that kind of information locally.
- The user revokes the presence sharing with a contact from one of their clients.
  Again his other clients that are online will receive unexpected presence notifications indicating that the subscription to the contact's social presence information was terminated. If they cache the information in the Shared XDMS's "*resource-lists*" document locally, they shall use this notification as a trigger to verify that the information is still up-to-date.
  Changes are done while the client was offline. A client that caches the information in the Shared XDMS's "*resource-lists*" document locally should check whether that document has changed when it comes online. Therefore, this will not cause any issues.
- The user owning multiple clients changes a contact from being a VIP contact to being non-VIP contact from one of their clients. His other clients that are online will receive unexpected presence notifications indicating that the subscription to the contact's social presence information was terminated. If they cache the information in the Shared XDMS's "*resource-lists*" document locally, they shall use this notification as a trigger to verify that the information is still up-to-date.
- The user owning multiple clients changes a contact from being a non-VIP contact to being a VIP contact from one of their clients. In this case, their other clients will receive presence notifications indicating that a subscription to the contact has been created and notifications including the other user's social presence information. Again, the clients shall use these unexpected notifications as triggers to update the locally stored copy of the Shared XDMS's "*resource-lists*" document if they cache that kind of information locally.

3.7.4.6 RLS Server Handling

3.7.4.6.1 Nickname Handling

A RLS server supporting RCS shall include any display name it received in the *P-Asserted-Identity* and *From* headers of the RLS subscription in the corresponding header of the related backend subscriptions that it sends to the Presence Server.

3.7.4.7 Presence Server Handling

3.7.4.7.1 Nickname Handling

A Presence Server supporting RCS shall include any display name it received in the *P-Asserted-Identity* header field of a presence subscription in the display-name attribute of any entry related to that subscription in the *application/watcherinfo+xml* body that is sent to the clients of the served RCS presentity that was the target of the subscription. If the *P-Asserted-Identity* header field does not contain any display name, the display name provided in the *From* header field of the subscription will be used, if any.

3.7.4.8 XDM Server Handling

3.7.4.8.1 Status Icon

In the network the retrieval of the information referred to by the link to the status icon will be realised in an architecture as described in [XDM1.1_AD] with the addition of the Cross-Network Proxies and XDM-8 and NNI-1 interfaces defined in [XDM2.0_AD]. The required functionality of the Cross-Network Proxy is limited to the authorisation, data transfer and routing of XCAP functionalities. The routing of search requests is not applicable to RCS. For RCS the supported protocols on the NNI-1 interface are limited to XCAP, "*limited XQuery over HTTP*" is not supported.

At the functionality level, this means that the identity provided by the Aggregation Proxy is not only shared on the XDM-4 and enabler specific reference points between the Aggregation Proxy and the Enabler specific XDMS as it is described in [XDM1.1_Core] section 6.4.1, but also on the XDM-8 and NNI-1 interfaces as it is described in [XDM2.0_Core] section 5.1.3. The Integrity and Confidentiality protection of [XDM1.1_Core] section 6.4.2 is extended to the NNI-1 interface as it is described in [XDM2.0_Core] section 5.1.4. Furthermore in addition to the functionality described in [XDM1.1_Core], the Aggregation Proxy shall route requests to the Cross-Network proxy as it is described in [XDM2.0_Core] section 6.3.1.1 and route the Cross-Network Proxy's responses back to the XDM client. The procedures for routing requests to the search proxy that are described in [XDM2.0_Core] section 6.3.1.1 are not applicable for RCS. Finally the functionality of the Cross-Network Proxy as it is described in [XDM2.0_Core] section 6.5 and subsections shall be supported with the exception of all functionality related to the routing of Search Requests and Search Responses.

3.7.5 NNI and IOT considerations

The NNI interfaces for SPI sharing shall behave according to the procedures described in section 2.12 and the documents it refers to.

3.7.6 Implementation guidelines and examples

3.7.6.1 SPI transaction handling

<u>Initiator side</u>

1. An RCS user that wants to Share SPI with a contact selects the contact entry in their local enriched address book.
2. They select in the menu “share” (if available, that is the contact has the SPI service capability) the function “Share Social Presence” and can see by using the SPI general menu the SPI status associated with the contact (“idle”, “pending” “activated”, “terminated”)
3. This SPI general menu, depending on the SPI status, enables them to invite the contact to share SPI with following options

    - “VIP contact”: YES / NO (default NO)
    - “Authorise Location Sharing”: YES / NO (default NO)

NOTE: at any moment, for these 2 options, when the SPI status becomes “active”, the general Share SPI menu offers the user the possibility to change their choices

    - “Nickname” text field: free user text

4. Then the user can follow the SPI status evolution the SPI status by selecting the contact and activating the SPI general menu

    - “pending”: the contact has not yet accepted to share SPI with them
    - “active”: the contact has accepted to share SPI with them
    - “terminated”: The contact, after acceptation, has decided to revoke sharing Presence Information

<u>Callee side</u>

1. The RCS user is triggered by a pop up SPI menu that a distant user has invited them to share their Social Presence Information

    - If the user already has a contact entry for the inviting user in the local address book, then the name assigned to the contact entry in the local address book of the user appears in this SPI menu
    - If the user is not present in the local address book, then the “nickname” of the inviting user (if any provided) and their E.164 address appear in the menu instead

2. The SPI pop up menu proposes allows actions through buttons and fields to be filled

    - “Accept”: YES/NO
    - “VIP contact”: YES / NO (default NO)
    - “Authorise Location Sharing”: YES / NO (default NO)

NOTE: at any moment, for the latter 2 options, when the SPI status becomes “active” the general Share SPI menu offers the user the possibility to change their choice

3. Then the user can follow the SPI status evolution by selecting the contact and activating the SPI general menu

    - “active”
    - “terminated”: The contact, after acceptation, has decided to revoke sharing Presence Information

SPI status “active”

At any moment, in the “active state” the user can choose for a contact selected in the address book:

- To modify SPI sharing parameters: VIP contact, Geolocation Sharing authorisation
- To revoke SPI sharing

3.7.6.2 Availability handling

The user can choose how they appear to their contacts: “Available” or “Not Available”.

3.7.6.3 Free Text handling

The user enters some free text possibly including emoticons. They are blocked when the length of the text reaches the limit fixed by the Service Provider.

3.7.6.4 Icon handling

The user is asked to choose an image in the local file system of the device from a sub set of the images that are candidate to be part of the user SPI (filter based on file size).

3.7.6.5 URL label

The user is asked to enter a URL and an associated free text. The user may be assisted by the application to enter the information.

3.7.6.6 Geolocation handling

In a manual mode, user manually picks a position (x, y) on a map or user requests for an update of their position (x, y) information. Then, geolocation information is given by RCS client towards authorised enriched contacts as soon as it has been made available on the RCS client by the user.

In automatic mode, update of location coordinate information (x, y) is automatically made and given to the authorised enriched contacts on a regular basis.

Manual mode and automatic mode are further detailed below.

3.7.6.6.1 Display Modes

Three displays modes are possible:

1. Text: a user is located and the result is given to their authorised enriched contacts under a declarative text format (Paris, La Défense). The declarative text is always manually edited by the user.
2. Map: a user is located and the result is given as coordinate information (x, y) to his authorised enriched contacts and displayed under a map format. When the user is displayed as a dot on a map, their location information can also be displayed as text

in other screens. For example, if a user has updated his location to a position in the centre of London on a map, some screens without a map may display his location using the declarative text edited by the user (for example, “London, UK”).
3. A combined display of text and a map

3.7.6.6.2 Update Information

Declarative location text information is always manually edited/updated by the user.

The Geolocation information update regarding coordinate information (x, y) can be either:

- Manual
    - The user can select their location manually on a map, by either entering text that is then processed to provide location (as coordinate information [x,y]) on a map (for example Google Maps) or, for example, by dragging and dropping a “pin” on a map to the desired location. This user-chosen location can be different from the user’s actual location.
    - Triggering their actual current location (based, for example, on a GPS signal from the device or a mobile network-based location). For example, they click on the location update button, and coordinate information (x,y) is automatically filled)
- Automatic
- (User A decides that they want their authorised contacts to be informed regarding their coordinate position (x,y) on a regular basis). Location coordinate information (x, y), and any update is automatically made and given to authorised enriched contacts on a regular basis.

Other recommendations for implementation from the end user’s perspective (these are only meant as examples and not actual specifications):

- For Fully Automatic update, the user shall be able to choose the level of accuracy for their location
    - Country
    - City
    - Street (most accurate location)
- In addition to having a map displayed per contact inside the address book (at -1 or -2 navigation levels), there might be the possibility to have a consolidated map with all contact location information (within the scope defined : country, city or street). The starting position of the map is the user’s current position, if available. See also section 3.10.

3.8 IP Voice Call

3.8.1 Feature description

This feature provides an IP Voice Call service on an RCS device. This IP Voice Call service is used:

- To provide voice calling functionality on secondary devices

- To provide voice calling functionality on primary devices that do not have cellular coverage
- To provide PS voice calling on primary devices in LTE or EPC-integrated Wi-Fi coverage (as [PRD-IR.92] and [PRD-IR.51] respectively)
- As a fallback from the IP Video Call service (see section 3.9) when video media is dropped

An IP Voice Call interoperates with other RCS devices including VoLTE/VoWiFi as defined in [PRD-IR.92] and [PRD-IR.51] and with CS/PSTN (Public Switched Telephone Network) voice calls. The voice call is provided via IP Voice when the access network allows it, and may be provided via CS voice when IP Voice is not available, depending on the device and network capabilities.

The minimum set of supplementary services provided is described in [PRD-IR.92].

At any time, either user can terminate the IP Voice Call.

3.8.2 Interaction with other RCS features

3.8.2.1 Interaction with RCS Messaging, File Transfer, Content Sharing, Geolocation PUSH

IP Voice Call must use a separate SIP session which is not shared with Standalone messaging (section 3.2), Chat (section 3.3), Group Chat (section 3.4), File Transfer (section 3.5), Content Sharing (section 3.6) or Geolocation PUSH (section 3.10). Interaction with Content Sharing is covered in section 3.6.2. Interaction with Video Call is covered in section 3.9.2.

3.8.2.2 Interaction with CS Voice (Telephony)

In RCS, voice services are supported both on access networks natively offering CS Voice telephony (thus without the need for CS/PS voice conversion by Media Gateway components in the core network) as well as on IP based access networks using PS voice.

Within the RCS multi-device context, any of the user’s clients could be receiving the final leg of the voice service in any of these ways:

- through the CS network as CS Voice telephony,
- Through the IMS network using PS voice.

For NNI, the interconnection between networks could use CS voice service NNI or could use PS voice. For both the UNI and NNI interfaces, several options are possible to achieve the required behaviour, and neither choice is optimal in all circumstances. Therefore, this choice is considered out of scope for RCS.

3.8.3 High Level Requirements

3-8-1 The scope of the requirements for IP Voice Call are those found in [PRD-IR.92] and [PRD-IR.51].

3.8.4 Technical Realisation

At a technical level the voice call service shall be based on [PRD-IR.92] and [PRD-IR.51] and may either be realised

- As described in [PRD-IR.92] and [PRD-IR.51] for primary devices enabled for VoLTE/VoWiFi in which case [PRD-NG.102] shall be supported as well. Or
- As described in section 3.8.4.1 for other devices that don't currently support CS voice calls (i.e. secondary devices and primary devices not enabled for VoLTE/VoWiFi without cellular access).

Since in RCS a user may register a primary and one or more secondary devices in IMS, incoming SIP requests are forked. This principle also applies to the case where the user has several SIMs assigned to the same phone number (i.e. the same IMS subscription), and consequently, incoming SIP requests are forked.

This also applies to incoming SIP requests for IP Voice Calls, so it is expected that they be forked in the same way as other RCS related SIP requests are forked, i.e. in parallel. For voice sessions set up according to [PRD-IR.92] and [PRD-IR.51], the support for early media as described in [PRD-IR.92] and [PRD-IR.51] is required.

Broadband Access clients which support and are configured for RCS IP Voice Call but are not enabled for VoLTE/VoWiFi (and therefore do not make use of the IMS APN as specified in section 2.9.1.4) shall behave as defined in section 3.8.4.1.

NOTE: When using the RCS IP Voice Call service, it is recommended that the device indicates to the user that this is not a telephony replacement service.

Furthermore the continuity of such an RCS IP Voice Call relies on the continuity of the IP connectivity (i.e. SR-VCC does not apply).

3.8.4.1 Devices using RCS IP voice calls

Table 50 summarises the sections in [PRD-IR.92] and [PRD-IR.51] that apply and do not apply to an RCS IP Voice Call, and where relevant provides a reference to the section where alternative procedures are found.

Document(s)	Relevant sections from [PRD-IR.92], [PRD-IR.51]	Applicability
[PRD-IR.92], [PRD-IR.51]	**All sections not mentioned below**	**Applicable**
[PRD-IR.92], [PRD-IR.51]	2.2.1 SIP Registration Procedures	The MMTEL IMS Communication Service Identifier (ICSI) shall be included in the Contact header field. According to the rules defined in section 2.4.3 *+g.gsma.rcs.telephony* is either also included in the Contact header field and filled with the values *cs* or *none* or it is not included in the *Contact* header field at all. In addition if RCS IP Voice Call is enabled but RCS IP Video Call is not enabled the client shall also include *+g.gsma.rcs.ipcall* in the *Contact* header field See also section 2.4

Document(s)	Relevant sections from [PRD-IR.92], [PRD-IR.51]	Applicability
[PRD-IR.92], [PRD-IR.51]	2.2.2 Authentication	Not applicable. See section 2.13
[PRD-IR.92], [PRD-IR.51],	2.2.4 Call Establishment and Termination	The client shall include the MMTEL ICSI in the *Contact* and *Accept-Contact* header fields for an RCS IP Voice/Video Call as per [PRD-IR.92] and [PRD-IR.94]. For an RCS IP Voice Call which can be upgraded to an RCS IP Video Call, the client shall include the MMTEL ICSI in both the *Contact* and *Accept-Contact* headers and the video tag in just the Contact header as per [PRD-IR.94].
[PRD-IR.51]	2.2.7 Hosted NAT Traversal	Not applicable. See section 2.8
[PRD-IR.92], [PRD-IR.51]	2.4.2 Integration of resource management and SIP	Applicable only if the IMS well-known APN is used for RCS (see section 2.9.1.4)
[PRD-IR.92], [PRD-IR.51]	2.5 SMS over IP	Not applicable
[PRD-IR.92]	3.2.6 Jitter Buffer Management Considerations	Not applicable
[PRD-IR.92]	3.2.7 Front End Handling	Not applicable
[PRD-IR.51]	4 Radio and packet core feature set	Entire section (including subsections) not applicable
[PRD-IR.92]	4.1 Robust Header Compression	Not applicable
[PRD-IR.92]	4.2 LTE Radio Capabilities	Not Applicable
[PRD-IR.92]	4.3 Bearer Management	Not Applicable
[PRD-IR.92]	4.4 P-CSCF Discovery	Not Applicable, see section 2.4.6
[PRD-IR.92], [PRD-IR.51]	5.1 IP Version	Not Applicable
[PRD-IR.92]	5.2 Emergency Service	Subject to local regulation
[PRD-IR.51]	5.3 Emergency Service	Subject to local regulation
[PRD-IR.92]	5.3 Roaming Considerations	Not Applicable
[PRD-IR.51]	5.4 Roaming Considerations	Not Applicable
[PRD-IR.92]	5.4 Accesses in addition to E-UTRAN	Not Applicable

Document(s)	Relevant sections from [PRD-IR.92], [PRD-IR.51]	Applicability
[PRD-IR.92]	Annex A: Complementing IMS with CS (A.1 General, A.2 Domain Selection, A.3 SR-VCC, A.4 IMS Voice service settings management when using CS access, A.5 Emergency Service, A.6 Roaming Considerations, A.7 SMS Support) In [PRD-IR.94] Annex A Complementing IMS with CS (A.1 General, A.2 SR-VCC)	Not Applicable

Table 50: IR.92 and IR.51 applicability to RCS IP Voice Call

A device/client supporting and configured to use RCS IP Voice Calls shall indicate this in SIP INVITE requests and responses according to Table 50. The device/client shall also include a *P-Preferred-Service* header field with the MMTEL ICSI as per [PRD-IR.92] and include the relevant subclass, i.e.

```
P-Preferred-Service: urn:urn-7:3gpp-service.ims.icsi.mmtel.gsma.ipcall
```

3.8.4.2 Flows

Since the voice call UX is well-known, it is not necessary to provide basic message flows and a reference UX. A flow is provided for handling of an incoming CS call when there is already an IP Voice Call.

3.8.4.2.1 Incoming CS Voice call when already in an RCS IP Voice Call

In this scenario an RCS IP Voice Call is ongoing between two users.

User A receives an incoming CS Voice call from User C. User A shall receive an indication of the incoming call. User A shall be able to:

1. Reject the incoming CS Voice call from User C (and thus stay in the same RCS IP Voice Call with User B as long as data connectivity was not lost);
2. Accept the incoming CS Voice call from User C, and consequently tear down the RCS IP Voice Call with User B;
3. Put the RCS IP Voice Call with User B on hold (as long as data connectivity was not lost) and answer the incoming CS Voice call from User C.

NOTE1: If the device is using LTE and the CS Fallback network is 2G without DTM support, the data connection is suspended and resumed after the CS call, so the RCS IP Voice Call may or may not still be there.

NOTE2: Given the use cases described in section 3.8.1, this situation can only occur for a device on which the (IP) voice call was established when the device was not in cellular coverage with the device entering such coverage during the call.

3.8.5 NNI and IOT considerations

No specific guidelines apply other than what is already defined in Section 2.12.

3.8.6 Implementation guidelines and examples

From the UX point of view, two possible entry points to the voice service are:

1. Address book/Call-log: A voice call can be initiated with any registered contact – contact oriented initiation.
2. Chat window: From the Chat (one-to-one Chat only) window a voice call can be initiated using the relevant menu item. The experience is identical to the address book/call-log.

Since the voice call UX is well-known, it is not necessary to provide implementation guidelines and examples.

3.9 IP Video Call

3.9.1 Feature description

This feature provides an IP Video Call between two RCS devices with synchronisation between the audio and video streams, thus providing lip synchronisation. For voice the IP Voice Call service (as described in section 3.8) is used, along with the clarifications for call establishment described in [PRD-IR.94].

The continuity of an RCS IP Video Call relies on the continuity of the IP connectivity (i.e. SR-VCC does not apply here).

The establishment of the IP Video Call session can be achieved in the following ways:

1. '**Direct launch**', if no previous voice call was established between the contacts.
2. '**Upgrade to IP Video Call'**, if the users were already engaged with each other in an IP Voice Call communication.

From the user experience perspective the RCS user can toggle between front camera ("me"), the rear camera ("what I see") and a file (video stream), at any time when using the IP Video Call service.

NOTE: The Video Call service in this context is seen as a superset of Video Share use cases as described in section 3.6.1.1 offering lip synch in addition.

In all cases and as described in [PRD-RCC.61], when invited for a video call an RCS user can either:

- Accept the video call establishing a full duplex video call
- Accept only to receive the inviting user's video content establishing a call where the video part runs in simplex mode alongside a full duplex audio call. In this case the accepting user can at any time decide to move the video part to full duplex as well.
- Accept the call as audio only, i.e. decline the video part of the communication. Voice call is established or continues.
- Decline the video call i.e. no communication is established to any of the receiving user's devices when declining the video call. The call may be redirected to a voice or video messaging system however depending on the policies of the receiving user's network.

When the video stream of the IP Video Call is realised in a full duplex mode, at any time, either user can decide to migrate from a full duplex mode to a simplex mode, i.e. deactivate the sending of their video stream. They can later decide to migrate from a simplex mode to a full duplex mode again.

At any time, either user can terminate the IP Video Call (both audio and video stream or only the video stream).

An RCS device may learn and remember that a contact is IP Video Call capable upon receiving a SIP INVITE request for an IP Voice or IP Video Call. Communication with [PRD-IR.94] compliant devices should not be prevented if RCS procedures for service capability discovery are not supported by those devices.

3.9.1.1 Direct Launch

When both parties support video call at any particular point in time (e.g. by the capability exchange described in section 2.6), either user can initiate the setup of a video call. The receiving user determines whether the call will be initiated in full or simplex mode.

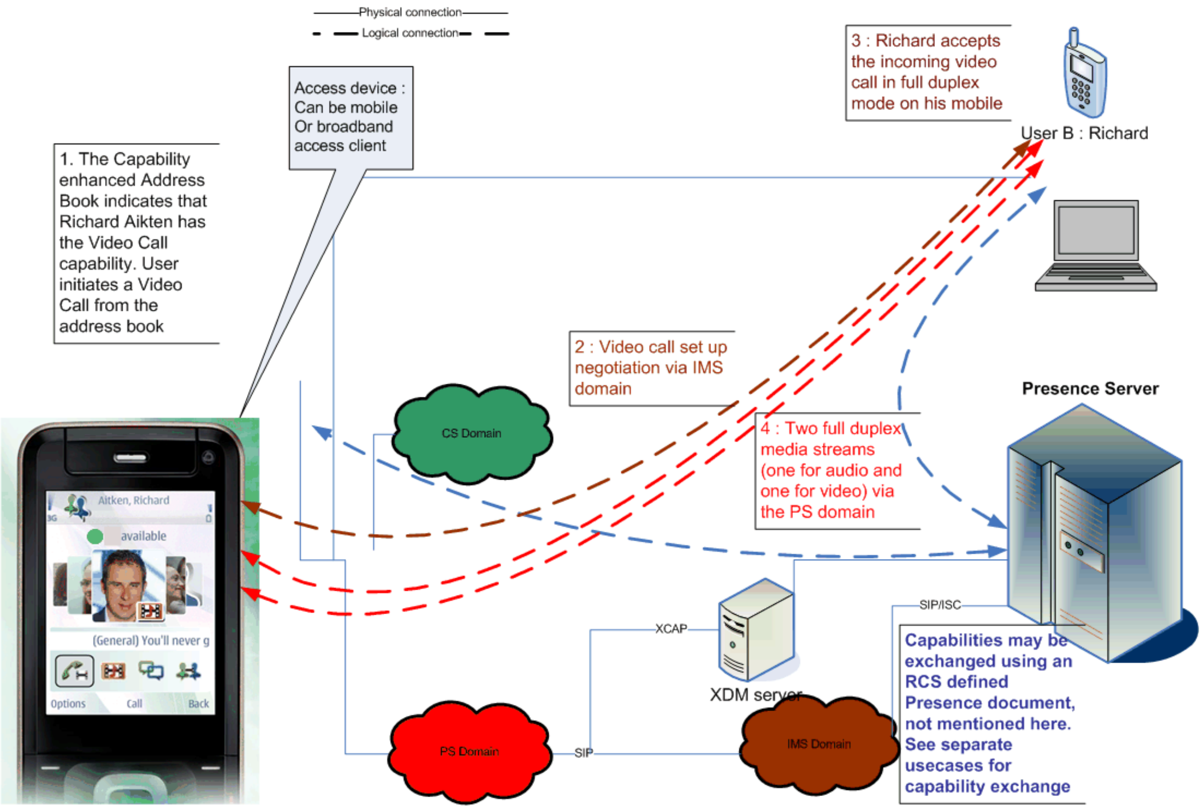

Figure 88: Full duplex video call

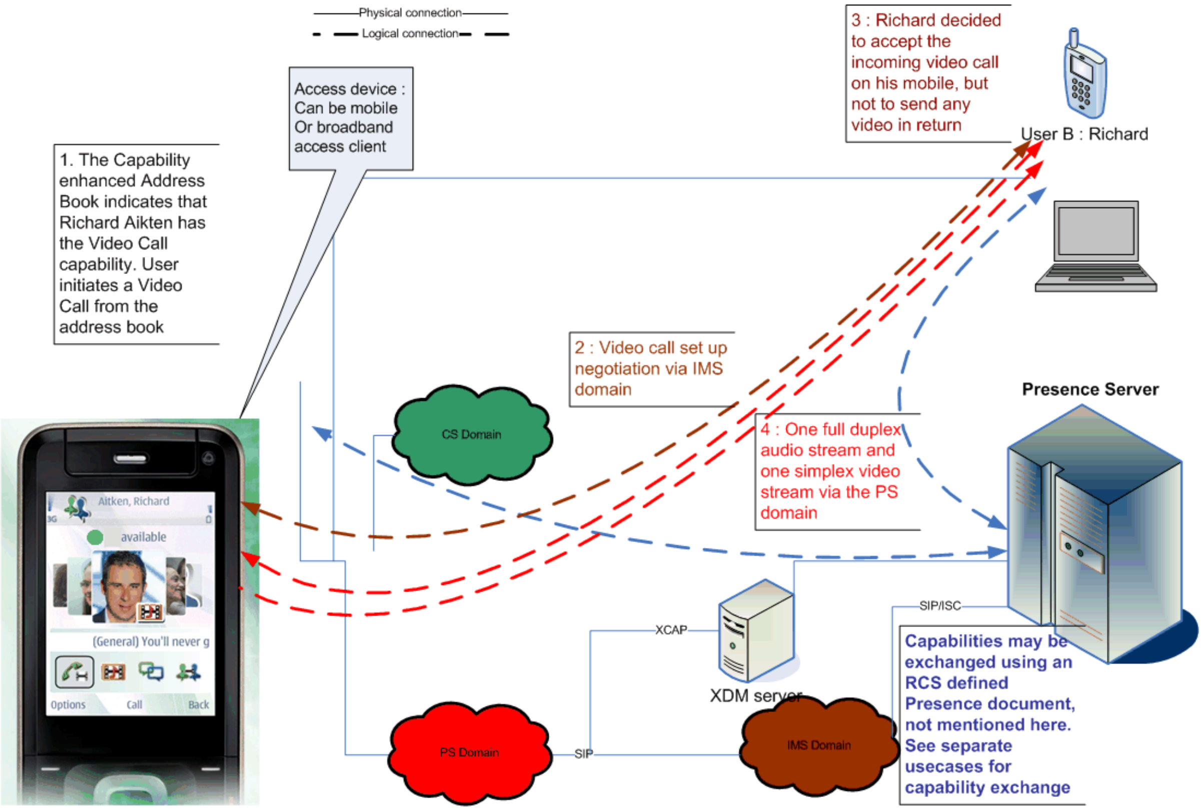

Figure 89: simplex video call

Users could switch between the full duplex and simplex variants of the video during the call. This would result in a new negotiation via the IMS domain for the ongoing call.

NOTE: multiparty calls are also possible.

3.9.1.2 Upgrade to IP Video Call

As stated in section 3.9.1, a user could also start a video call from an existing end-to-end IP Voice Call (that is the service described in section 3.8). When one of the parties in the voice call is having the voice call as a CS voice call, the upgrade to an IP Video Call shall not be offered to the user. If Video Share is supported, upgrade to Video Share may still be possible.

When the devices on the call all support video call at a particular point in time, either user can initiate the upgrade to a video call by selecting the corresponding option.

If the voice call was entirely (end-to-end) in the PS domain this initiates a negotiation via the IMS domain and if the other user accepts the upgrade a simplex or duplex video stream is added to the ongoing call.

NOTE1: If one end of the call moved to CS, the upgrade may fail, but the voice call would remain in place. If the other party is an RCS user, the party wanting to upgrade may have discovered the fact that video call is no longer available due to the capability exchange described in section 2.6 and should therefore not be offered the possibility to upgrade to an IP Video Call.

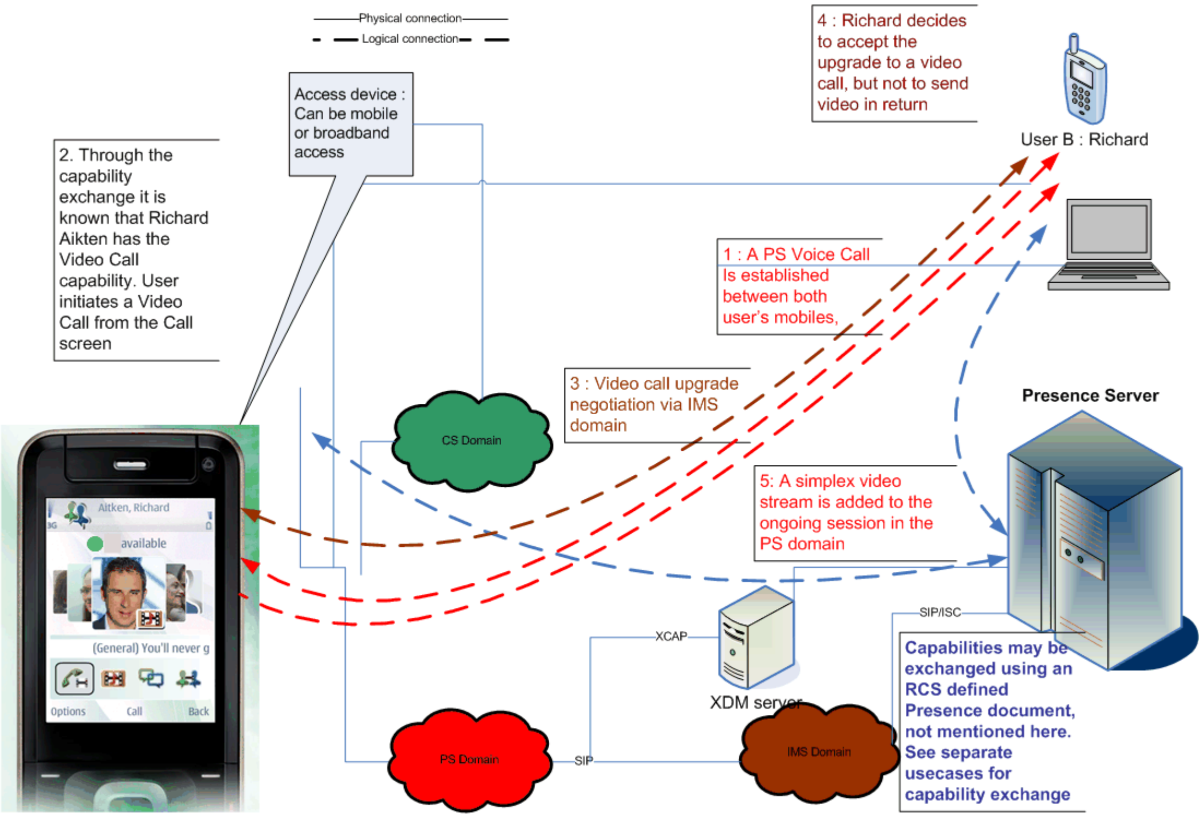

Figure 90: Upgrade PS call to video call

NOTE2: The behaviour is the same for the scenario where the user accepted the video call as a full duplex service.

3.9.2 Interaction with other RCS features

3.9.2.1 IP Voice Call

The IP Video call must use the same SIP session as the IP Voice Call (see section 3.8).

The video call service has a strong interaction with the voice call service since both services offer the option for full-duplex real-time communication. That strong relation results in the option to upgrade an existing voice call to a video call as described in section 3.9.1.2. An end-to-end IP Voice Call is upgraded by adding an additional media stream to the ongoing session.

Communication Waiting: when the user is on a voice call and a request for an unrelated video call is received (or vice versa), the device shall handle this video call in the same way as a second voice call coming in. Therefore it will behave differently from the scenario where no call was active and will thus not start ringing loudly and shall use Communication Hold appropriately if the new call is accepted without terminating the ongoing one.

3.9.2.2 Video Share

The IP Video Call and Video Share service capabilities are mutually exclusive: when both ends are capable of using the IP Video Call service (as per [PRD-IR.94]), then IP Video Call shall be used as the service to share contents instead of Video Share as described in section 3.6. If one or both ends are not capable of using the IP Video Call service, then Video Share will be used to provide the service. Therefore when performing a capability exchange within a call, if the Video Call capability is set as available, the Video Share capability shall also be made available.

3.9.3 High Level Requirements

3-9-1 In a video call the delay difference between audio and video media shall be unnoticeable (that is lip sync is provided)
3-9-2 The overall delay on both media shall allow for a conversational service
3-9-3 The quality of the video shall be high. At least H.264 level 1.2 shall be supported in suitable circumstances matching the similar requirement in [PRD-IR.94]
3-9-4 It shall be possible to establish a video call without having an active voice call between the parties in the call
3-9-5 It shall be possible to convert an ongoing IP Voice Call (that is as in section 3.8) into an IP Video Call
3-9-6 The receiver shall be able to accept the call in full-duplex mode and in simplex mode in which case no content is sent back to the originating party.
3-9-7 It shall be possible for either party to turn a full duplex video call into a simplex one by terminating the streaming.
3-9-8 If the device has multiple cameras it shall be possible to toggle between them.
3-9-9 The receiver shall be able to reject the video call. This rejection does not affect an ongoing voice call.
3-9-10 Either party shall be able to terminate an active video call
3-9-11 Terminating an active video call shall terminate the communication regardless of whether the call was initiated directly as a video call or initially started as a voice call only.
3-9-12 At least the minimum set of supplementary services defined in [PRD-IR.94] shall be supported

3.9.4 Technical Realisation

Depending on the client configuration and client limitations, two technical enablers to realise the IP Video Call service may be available:

- ViLTE (Video over LTE) as defined in [PRD-IR.94], and,
- RCS IP Video Call as described in section 3.9.4.1

A device enabled for VoLTE/VoWiFi shall offer IP Video Call according to [PRD-IR.94] and [PRD-IR.51]. Integration of resource management and SIP is done as per [PRD-IR.94] for devices currently supporting VoLTE, and as per [PRD-IR.94] and [PRD-IR.51] for devices currently supporting VoWiFi.

For other devices (i.e. primary devices not enabled for VoLTE/VoWiFi or secondary devices), the device shall offer IP Video Call as an RCS IP Video Call according to section 3.9.4.1. This would mean providing this service on a best effort basis, the device shall clearly indicate to the user that they are not using a telephony replacement service. For such devices the service will only be available depending on the Service Provider policy settings (*PROVIDE RCS IP VIDEO CALL* as defined in section A.1.12). For such devices, no specific requirements for resource management are required.

For RTP media and RTCP usage, a device using non-cellular access shall follow the requirements for NAT traversal as specified in section 2.8.

3.9.4.1 Devices using RCS IP video calls

Table 51 summarizes the sections in [PRD-IR.92] and [PRD-IR.51] that apply and do not apply to an RCS IP Voice Call or RCS IP Video Call, and where relevant provides a reference to the section where alternative procedures are found.

Document(s)	Relevant sections from [PRD-IR.92], [PRD-IR.51], [PRD-IR.94]	Applicability
[PRD-IR.94]	**All sections not mentioned below**	**Applicable**
[PRD-IR.92], [PRD-IR.51]	**All sections not mentioned below that are referred to from [PRD-IR.94]**	**Applicable**
[PRD-IR.92], [PRD-IR.51], [PRD-IR.94]	2.2.1 SIP Registration Procedures	The MMTEL IMS Communication Service Identifier (ICSI) shall be included in the Contact header field. According to the rules defined in section 2.4.3 *+g.gsma.rcs.telephony* is either also included in the Contact header field and filled with the values *cs* or *none* or it is not included in the *Contact* header field at all. In addition, • If RCS IP Video Call is enabled the client shall also include *+g.gsma.rcs.ipcall;video* in the *Contact* header field. • If RCS IP Video Call is enabled and the client supports the specific behaviour when receiving an RCS IP Video Call that cannot be downgraded by the user into an RCS IP Voice Call (see section 3.9.4.1.1) the client shall also include *+g.gsma.rcs.ipcall;+g.gsma.rcs.ipvideocallonly;video* in the *Contact* header field. See also section 2.4
[PRD-IR.92], [PRD-IR.51]	2.2.2 Authentication	Not applicable. See section 2.13

Document(s)	Relevant sections from [PRD-IR.92], [PRD-IR.51], [PRD-IR.94]	Applicability
[PRD-IR.92], [PRD-IR.51], [PRD-IR.94]	2.2.4 Call Establishment and Termination (2.2.2 Call Establishment and Termination in [PRD-IR.94])	The client shall include the MMTEL ICSI in the *Contact* and *Accept-Contact* header fields for an RCS IP Voice/Video Call as per [PRD-IR.92] and [PRD-IR.94]. For an RCS IP Voice Call which can be upgraded to an RCS IP Video Call, the client shall include the MMTEL ICSI in both the *Contact* and *Accept-Contact* headers and the video tag in just the Contact header as per [PRD-IR.94]. In addition to the above, • For an RCS IP Video Call the client shall also include *+g.gsma.rcs.ipcall;video* in the Contact header field. • For an RCS IP Video Call where video media cannot be removed by the user the client shall also include *+g.gsma.rcs.ipcall*;*+g.gsma.rcs.ipvideocallonly;video* in the *Contact* header field.
[PRD-IR.51]	2.2.7 Hosted NAT Traversal	Not applicable. See section 2.8
[PRD-IR.94]	2.4.1 Integration of resource management and SIP	If the video media stream is not providing for a sufficient Quality of Service (QoS) level, then the UE may, based on its preferences, modify, reject or terminate the SIP session, according to section 6.1.1 in 3GPP TS 24.229.
[PRD-IR.92]	3.2.6 Jitter Buffer Management Considerations	Not applicable
[PRD-IR.92]	3.2.7 Front End Handling	Not applicable
[PRD-IR.51]	4 Radio and packet core feature set	Entire section (including subsections) not applicable
[PRD-IR.92]	4.1 Robust Header Compression	Not applicable
[PRD-IR.94]	4.2 EPS Bearer Considerations for Video	Not Applicable
[PRD-IR.94]	4.3 LTE Radio Capabilities	Not Applicable
[PRD-IR.94]	4.4 HSPA Radio Capabilities	Not Applicable
[PRD-IR.92]	4.4 P-CSCF Discovery	Not Applicable, see section 2.4.6
[PRD-IR.92], [PRD-IR.51]	5.1 IP Version	Not Applicable

Document(s)	Relevant sections from [PRD-IR.92], [PRD-IR.51], [PRD-IR.94]	Applicability
[PRD-IR.92]	5.2 Emergency Service	Subject to local regulation
[PRD-IR.51]	5.3 Emergency Service	Subject to local regulation
[PRD-IR.92]	5.3 Roaming Considerations	Not Applicable
[PRD-IR.51]	5.4 Roaming Considerations	Not Applicable
[PRD-IR.92]	5.4 Accesses in addition to E-UTRAN	Not Applicable

Table 51: IR.92, IR.94 and IR.51 applicability to RCS IP Video Call

A device/client supporting and configured to use RCS IP video calls shall indicate this according to Table 51.

The device/client shall also include a P-Preferred-Service header with the MMTEL ICSI as per [PRD-IR.94] and include the relevant subclass, i.e.

```
P-Preferred-Service: urn:urn-7:3gpp-service.ims.icsi.mmtel.gsma.ipcall
```

3.9.4.1.1 IP Video Calls when IP Voice Calls are not supported

If due to configuration (i.e. the values of the PROVIDE RCS IP VOICE CALL and PROVIDE RCS IP VIDEO CALL defined in section A.1.12) a client supports an RCS IP Video Call but does not support user-switch to RCS IP Voice calls, the client shall not accept IP Calls that do not include video media in the SDP offer; however the client shall allow video to be removed from an ongoing RCS IP Video Call if the video is removed by the remote peer. It shall include the *+g.gsma.rcs.ipvideocallonly* feature tag in the Contact header field of the SIP INVITE requests and 200 OK responses that it sends for RCS IP Video Calls.

Similarly if a network element in the path between two clients allows for RCS IP Video Calls and not for RCS IP Voice Calls establishment (e.g. to enforce the interworking agreement for a particular NNI), that network element shall ensure that this restriction is reflected in the exchanged capabilities and include the *+g.gsma.rcs.ipvideocallonly* feature tag in the Contact header field of the SIP INVITE requests and 200 OK responses that are exchanged between the clients for RCS IP Video Calls. The network element shall then also ensure that an RCS IP Call is torn down or rejected if the SDP offer or answer does not include a video media stream.

If a client supporting RCS IP Video Calls receives the *+g.gsma.rcs.ipvideocallonly* feature tag in the Contact header field of respectively the SIP INVITE request or 200 OK response for an RCS IP Video Call, it should not modify the session removing the video stream (i.e. the video media line in the SDP) during an ongoing RCS IP Video Call or not remove the video media line in the SDP answer in case of the recipient. The client supporting RCS IP Video Calls may offer the option to turn the video stream into a uni-directional stream.

3.9.4.2 Flows

3.9.4.2.1 Assumptions

The following sections describe the relevant message flows and reference UX. Please note that the following assumptions have been made:

- For simplicity, the internal mobile network interactions are omitted in the diagrams shown in the following sections.
- For simplicity RTCP exchanges are omitted in the diagrams. They should be executed as described in [PRD-IR.94] and section 2.8
- The terminal comes with a front and rear camera. If one or both are missing, the user should be notified only with the available options.
- The capability exchange was performed already (as described in section 2.6). Both users are thus aware that the other party supports IP Voice and Video calls.

3.9.4.2.2 Direct Launch

3.9.4.2.2.1 Accept as bidirectional

In this scenario no voice call is ongoing between the users and User A decides to initiate a video call with User B. User B accepts the call as a fully bidirectional video call. This results in two bidirectional RTP/RTCP streams, one for the audio and one for the video.

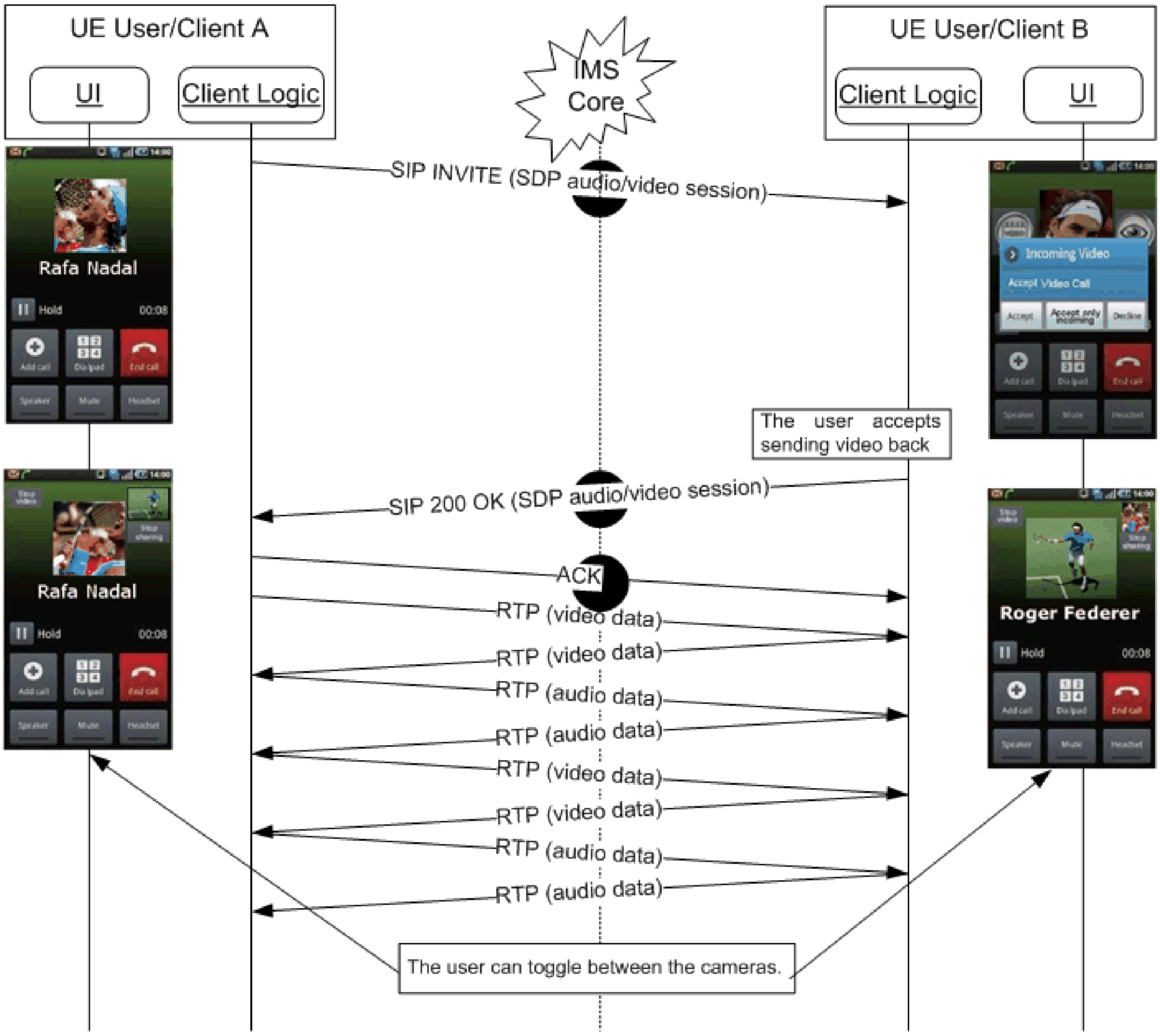

Figure 91: Direct launch of video call - Accept as bidirectional

3.9.4.2.2.2 Accept unidirectional

In this scenario no voice call is ongoing between the users and User A decides to initiate a video call with User B. User B accepts the call, but indicates that they do not want to send video back. This results in two RTP/RTCP streams, one bidirectional for the audio and one unidirectional (from User A to User B) for the video.

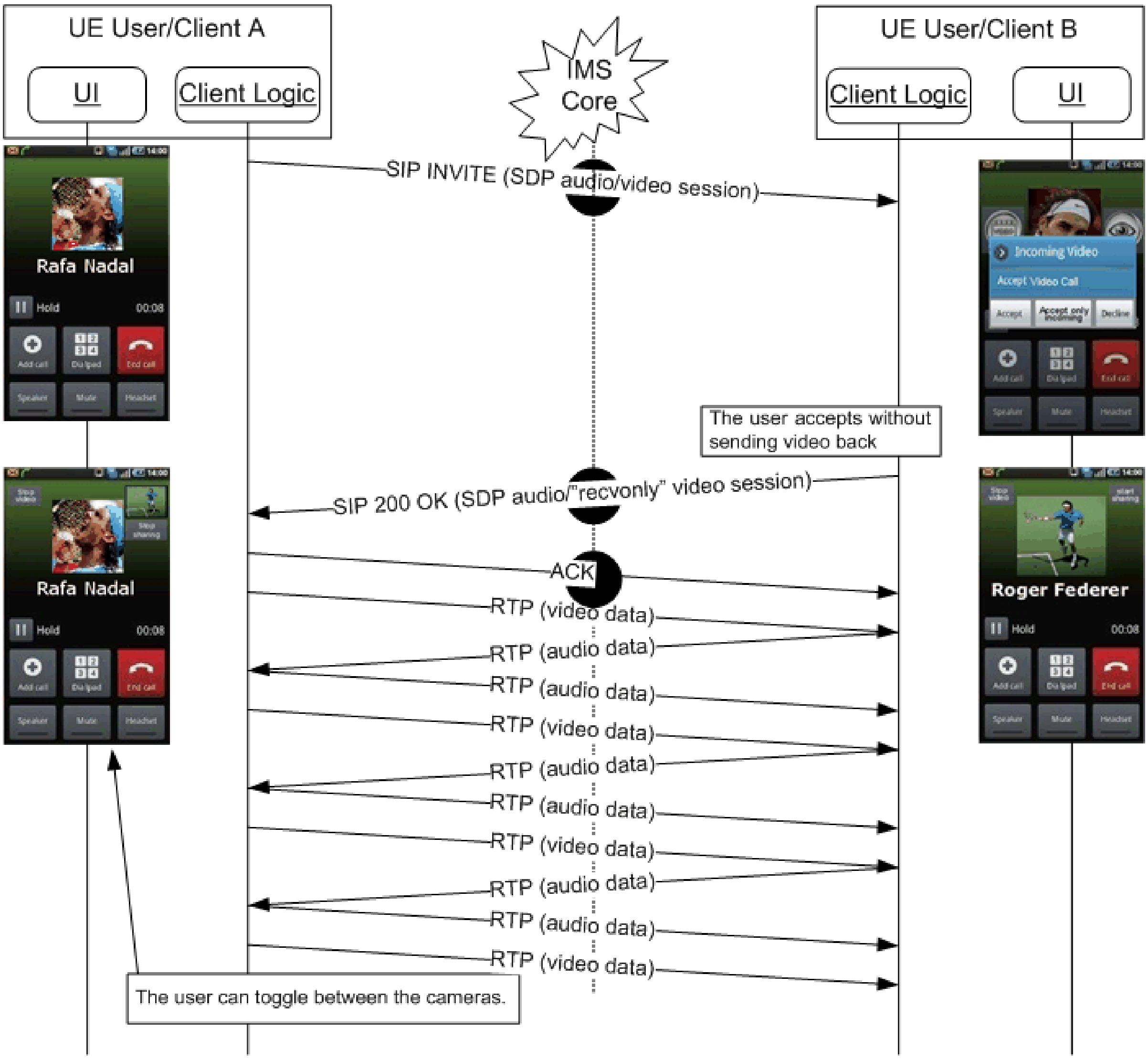

Figure 92: Direct launch of video call - Accept as unidirectional

3.9.4.2.2.3 Decline

In this scenario no voice call is ongoing between the users and User A decides to initiate a video call with User B. User B rejects the call.

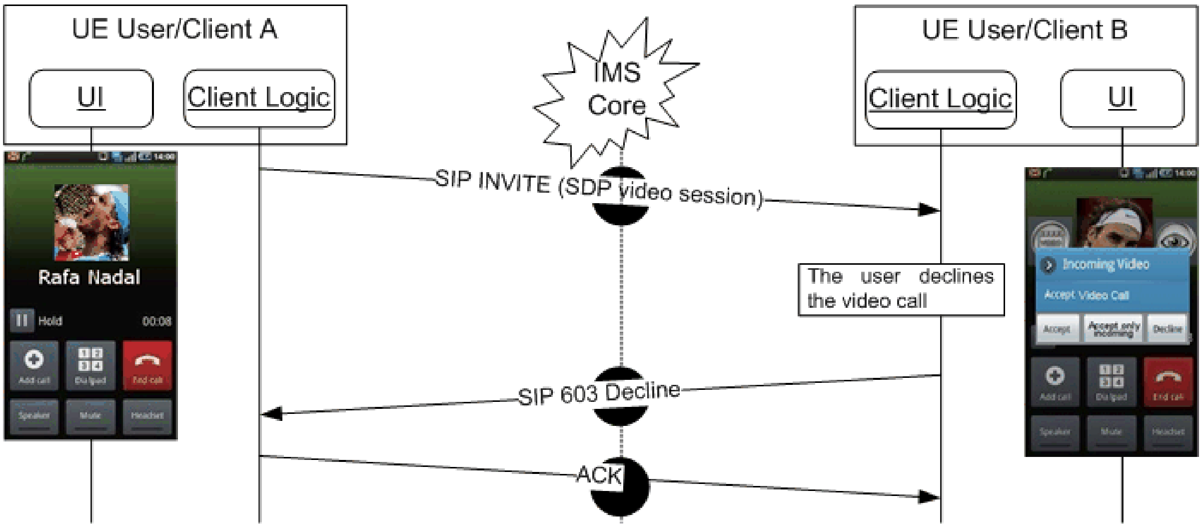

Figure 93: Direct launch of video call – Decline

In this scenario User B's network could also redirect the call to an announcement or voice/video mail system.

3.9.4.2.3 Upgrade from PS Call

3.9.4.2.3.1 Accept

In this scenario a PS voice call is ongoing between the users as specified in section 3.8. As specified in [PRD-IR.94] at the start of this call both terminals have indicated that they are capable of upgrading to a video call and no further capability exchange was done after the call setup indicating that this capability is no longer available.

User A decides to upgrade the ongoing call into a video call. User B accepts the upgrade (and in the illustrated flows decides to send video back). This results in a second RTP/RTCP stream for the video being added to the ongoing call (next to the existing bidirectional audio stream). This video stream can either be bidirectional or unidirectional depending on whether User B accepted to send video back or not. This is similar to the cases illustrated in sections 3.9.4.2.2.1 and 3.9.4.2.2.2.

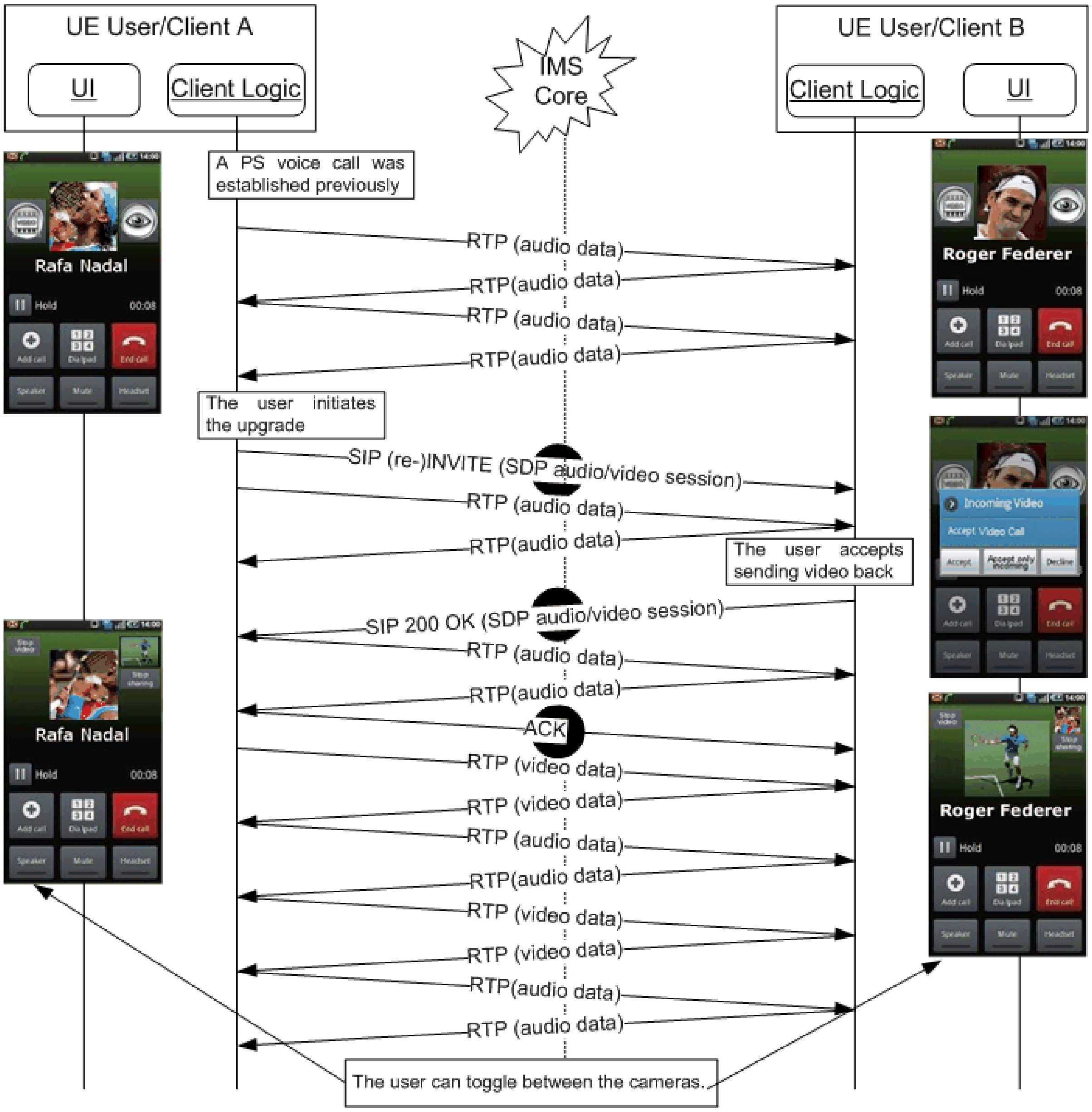

Figure 94: Upgrade PS Voice call to video call

NOTE: in a multidevice scenario the devices from User B that are not involved in the voice call will not be included in this upgrade flow.

3.9.4.2.3.2 Decline

In this scenario a PS voice call is ongoing between the users as specified in section 3.8. As specified in [PRD-IR.94] at the start of this call both terminals have indicated that they are capable of upgrading to a video call and no further capability exchange was done after the call setup indicating that this capability is no longer available.

User A decides to upgrade the ongoing call into a video call. User B declines the upgrade. The voice call continues unaffected.

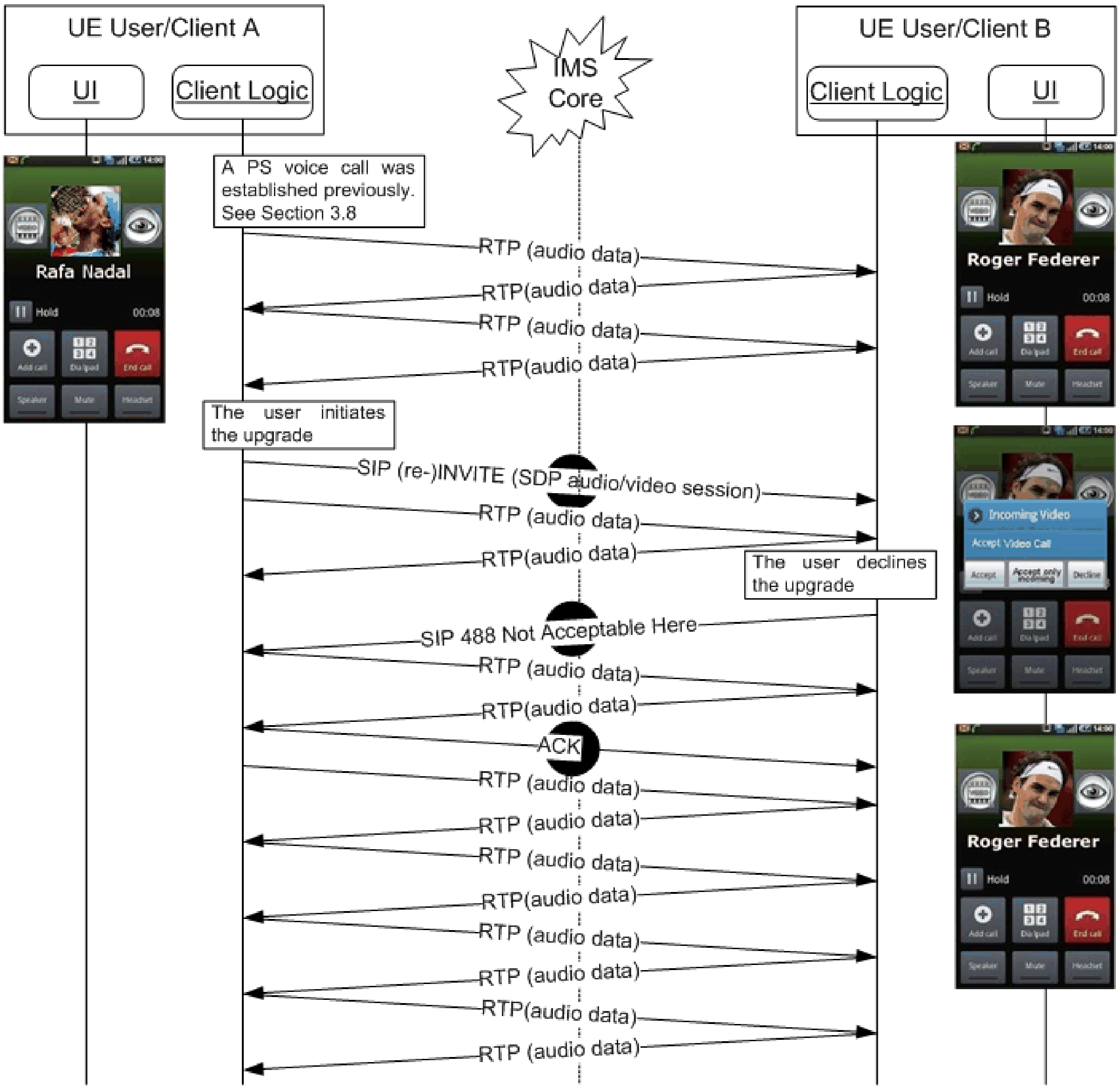

Figure 95: Decline upgrade PS Voice call to video call

3.9.4.2.4 Switch from unidirectional to bidirectional video

In this scenario User A and User B are involved in a video call in which User B is not sending video to User A. Then User B decides to start sending a video stream to User A.

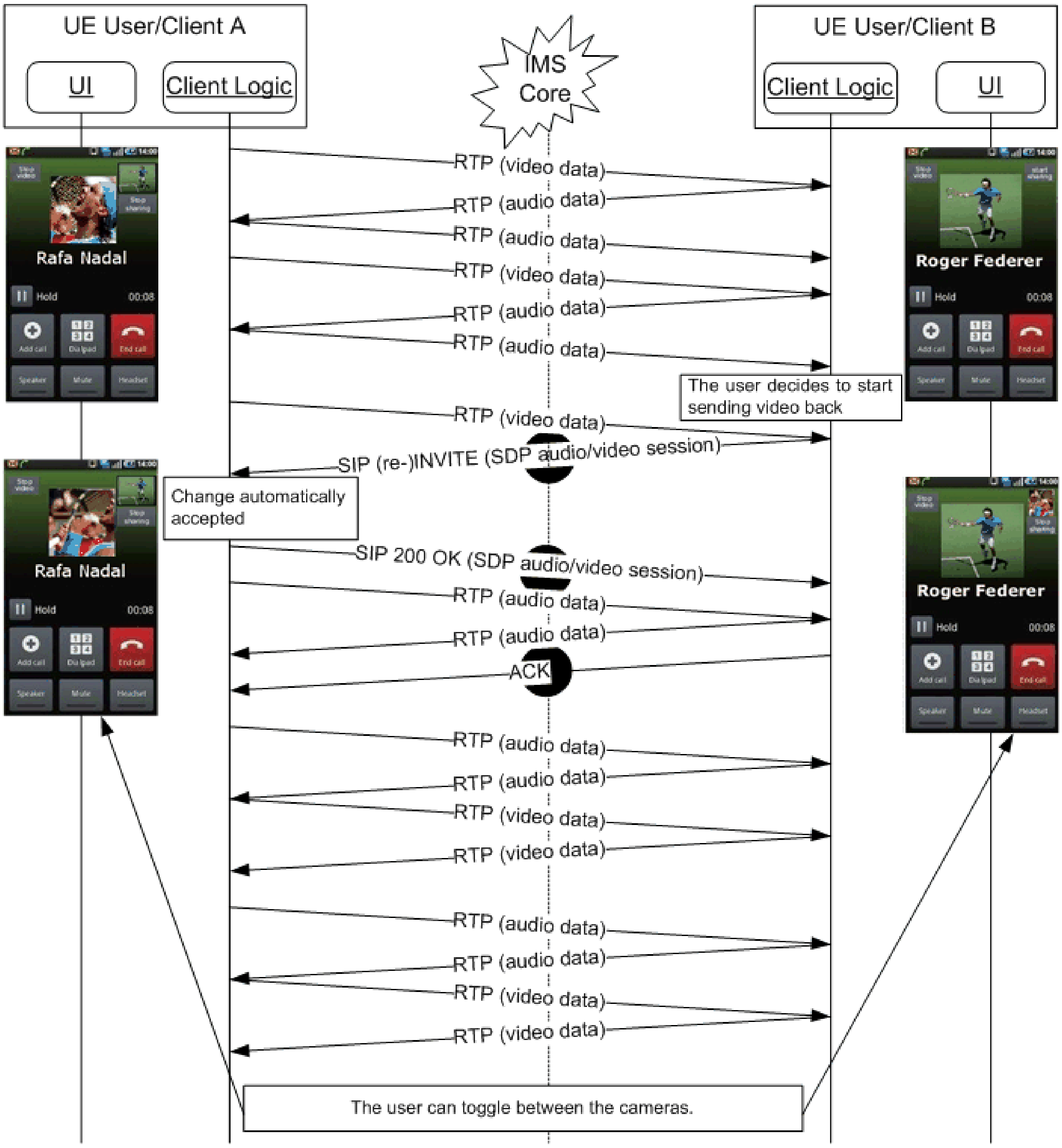

Figure 96: Change from unidirectional video call to bidirectional video call

3.9.4.2.5 Switch from bidirectional to unidirectional video

In this scenario User A and User B are involved in a video call in which both users are sending video to each other. Then User B decides to stop sending a video stream to User A.

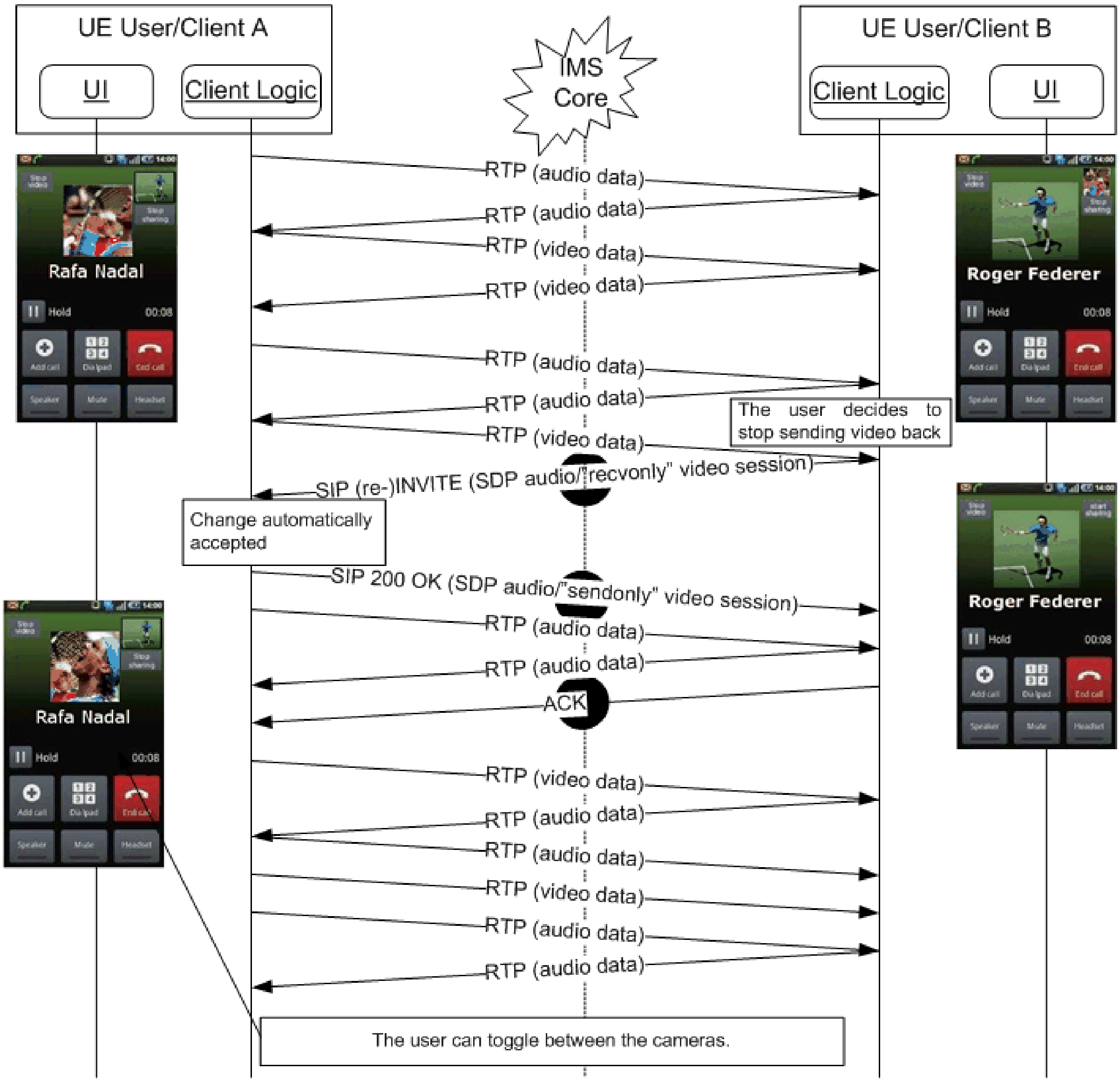

Figure 97: Change from bidirectional video call to unidirectional video call

3.9.4.2.6 Video call termination

In this scenario User A and User B are involved in a video call with each other and User A decides to terminate the call.

NOTE1: In this scenario User A is not necessarily the user that started the call.

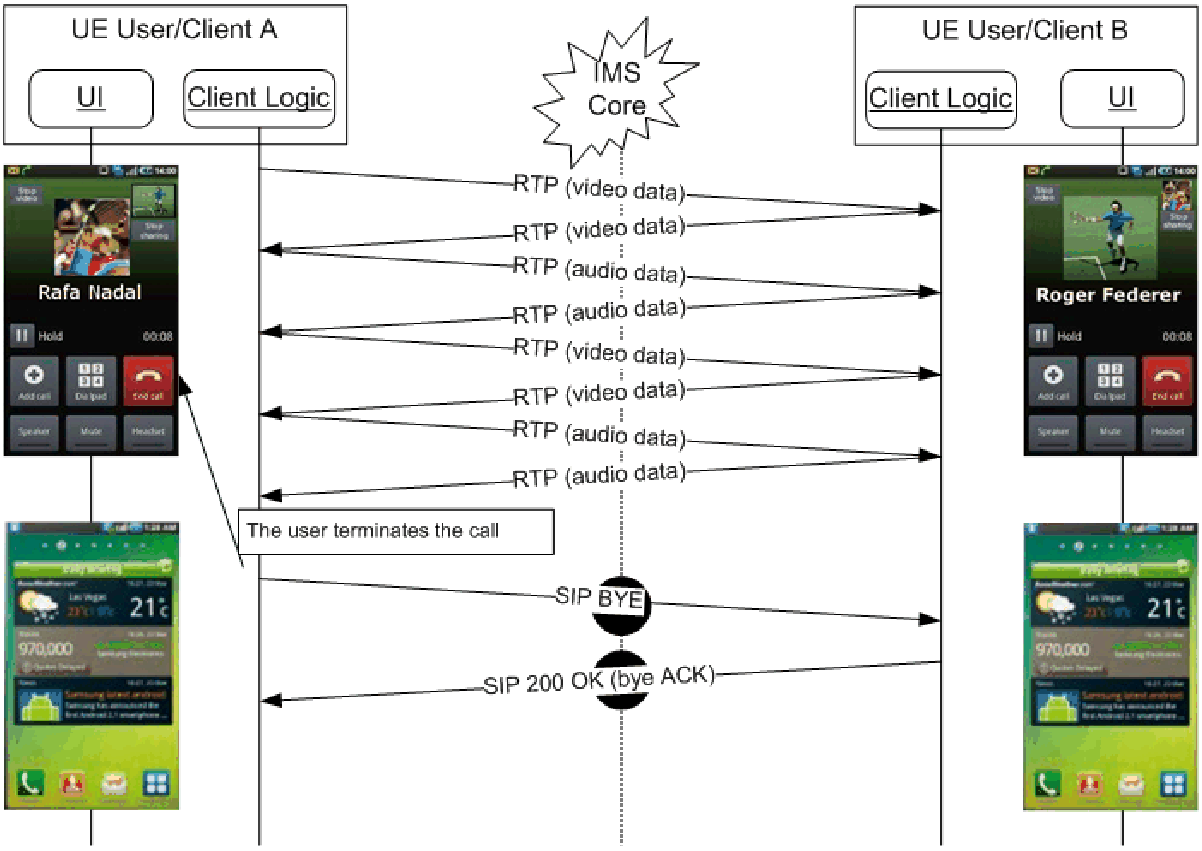

Figure 98: Video call termination

NOTE2: As this terminates the communication between User A and B, there is no need to do a capability exchange to verify whether the termination was or was not voluntary in contrast to the situation for Video Share described in section 3.6.4.

3.9.4.2.7 Incoming CS Voice call when already in an IP Video Call

In this scenario an RCS IP Video Call is ongoing between two users.

User A receives an incoming CS Voice call from User C. User A shall receive the announcement of the incoming call. User A shall be able to:

1. Reject the incoming CS Voice call from User C (and thus stay in the same IP Video Call with User B as long as data connectivity was not lost);
2. Accept the incoming CS Voice call from User C, and consequently tear down the IP Video Call with User B;
3. Put the IP Video Call with User B on hold (as long as data connectivity was not lost) and answer the incoming CS Voice call from User C.

NOTE1: If the device is using LTE and the CS Fallback network is 2G, the data connection is suspended and resumed after the CS call, so the IP Video Call may or may not still be there.

NOTE2: When a Service Provider's deployment allows directing incoming RCS IP Video Calls to devices that are in a CS voice call already, equivalent options will be available to the user unless the IP Video Call is originated by the same user as the conversation partner in the CS voice call. In that case, the incoming invitation for an IP Video Call should be presented to the user as the possibility to accept or deny the upgrade of the voice call to an IP Video call which also means that option 3 would not be available.

3.9.5 NNI and IOT considerations

The NNI interfaces for content sharing services shall behave according to the procedures described in section 2.12 and the documents it refers to.

3.9.6 Implementation guidelines and examples

From the UX perspective, there are three possible entry points to these services:

1. Address book/Call-log: A video call can be initiated with any registered contact providing the right capabilities are in place – contact oriented initiation.

Figure 99: User experience when starting from address book

2. Chat window: From the Chat (one-to-one Chat only) window a video call can be initiated using the relevant menu item. The experience is identical to the address book/call-log.
   The capability query is initiated when the user opens up the menu in which the available communication options are offered

Figure 100: User experience when starting from chat

3. <u>Call screen</u>: an ongoing voice call can be upgraded to a video call.

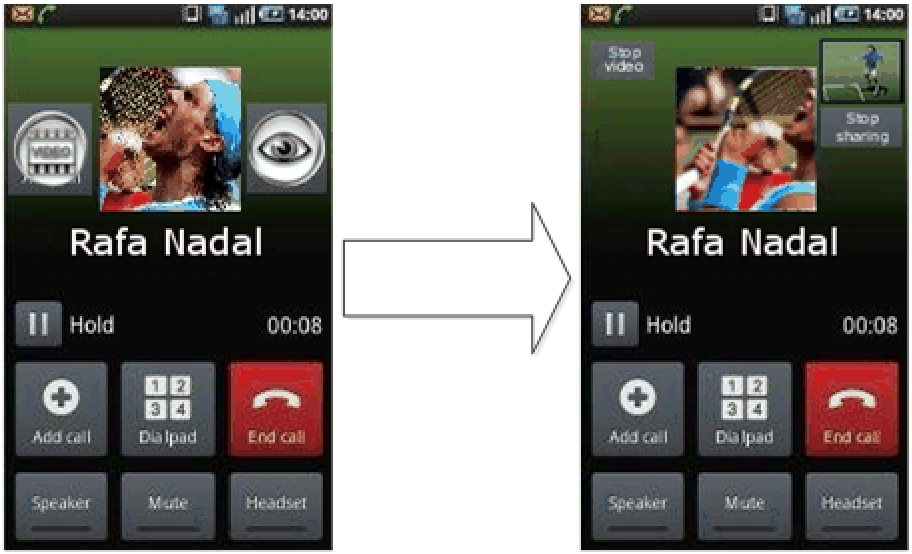

Figure 101: User experience when starting from call screen

Regardless of whether it is an upgrade scenario or a direct call, the receiver will always get 3 options on an incoming video call:

1. Accept
2. Accept only to receive video
3. Decline

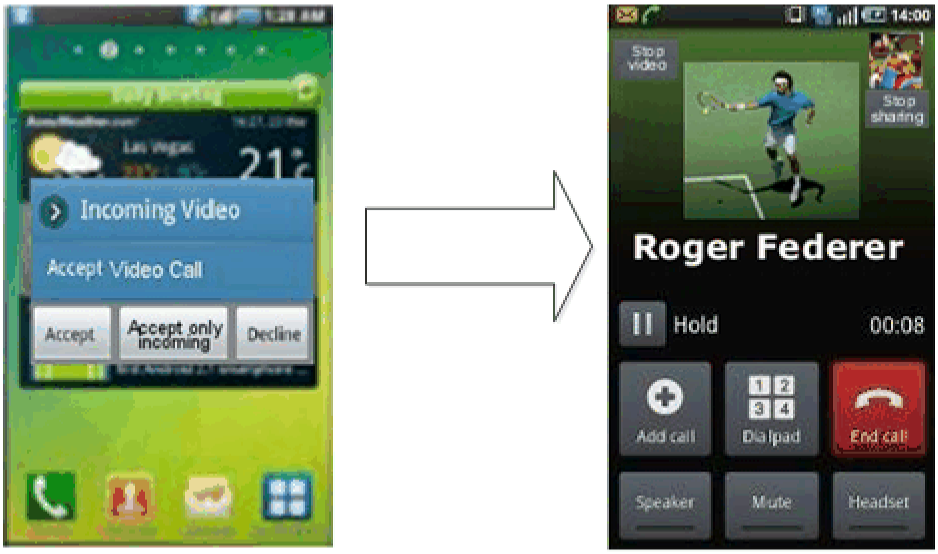

Figure 102: Video call receiver user experience direct video call

Figure 103: Video call receiver user experience – upgrade from voice call

3.9.6.1 Multidevice handling

When receiving an incoming IP Voice Call with video capabilities indicated as specified in [PRD-IR.94], it is recommended to have the recipient's devices supporting the IP Video Call display a video upgrade indication while it is alerting in order to draw the user's attention to the fact that answering at that device will allow the possibility to upgrade to a video call during the voice call.

3.10 Geolocation services

3.10.1 Feature description

Geolocation services comprise the following 2 features:

1. The "Geolocation PUSH" service that allows an RCS user to push location information (that can be the user location or the location of a suggested meeting point) to another RCS user
2. The "Geolocation PULL" service that allows an RCS user to retrieve the location information about another RCS user

It should be noted that similar services can be provided through the SPI with geolocation presence information (see section 3.7).

Their introduction in RCS is justified by the fact that an RCS user can have an interest to share geolocation information when SPI geolocation information cannot be used:

- Because SPI service is not offered by the Service Provider (if the 2 users belong to the same Service Provider) or one of the 2 Service Providers (if the 2 users do not belong to the same Service Provider)
- Because SPI is offered by the Service Provider (or the 2 Service Providers if the 2 users do not belong to the same Service Provider), but the 2 users do not want to share social information

3.10.1.1 Geolocation PUSH feature

Locations can be selected by the sender as follows:

- push current location
- push pre-defined location (e.g., the home address, a tool which permits a user to select from ‘favourite locations’ may be provided)
- push a location that is selected on a map

The user can also choose to put additional text information about the location.

3.10.1.2 Geolocation PULL feature

This feature is used by an RCS user, the origin RCS user, to retrieve the location information on any other RCS user – i.e. not limiting to users that share SPI with the RCS user

Behaviour at the origin RCS user side:

- When successful, the RCS user is informed with the result: geolocation coordinates (x, y).
- The user can then choose to store the information in the address book or/and show the information on a map

Behaviour at the target RCS user side

- The target user is informed that another RCS user is requesting to retrieve their geolocation
- The target user either authorises (ALLOW) or refuses (DENY) to share their geolocation
- If the target authorises (ALLOW) sharing their location, the location is retrieved automatically by the client/device accessing the Location Based Services (LBS) infrastructure in the network.

Multi device handling for the Geolocation PULL feature:

- The primary device will be the default recipient of the authorisation request. If the user replies ‘ALLOW’, this primary device will provide the user location information

3.10.2 Interaction with other RCS features

3.10.2.1 Geolocation PULL service

Interaction with RCS chat and voice/video call: the feature can be activated in the context of an established voice/video call (single point or multipoint) or in the context of an established RCS chat.

3.10.2.2 Geolocation PUSH service

Interaction with RCS chat and voice/video call: the feature can be activated in the context of an established voice/video call (single point or multipoint) or in the context of an established RCS chat.

The Geolocation PUSH service can also be used in the context of a 1-to-1 Chat, a Group Chat or a Call to deliver the "Show on a Map" functionality.

3.10.3 High Level Requirements

3.10.3.1 Geolocation PUSH

3-10-1 Geolocation information should be made available to any user (notwithstanding whether at home-PLMN or roaming in visiting-PLMN)
3-10-2 Shall be deployed as point to point service between 2 RCS users having the capability
3-10-3 An RCS user shall have the possibility to communicate geolocation information to a contact that has Geolocation PUSH capability
3-10-4 The service can be accessed from the address book or share menu
3-10-5 The service can be accessed also within a call, a chat or a Group Chat
3-10-6 Geolocation information shall consist of:

- Free text entered by the RCS user (optional)
- coordinates (x,y) (mandatory)

3-10-7 Coordinates can be obtained Manually

- The user referring to a predefined stored location
- Or the user picks the location point on a map.

3-10-8 Coordinates can be obtained Automatically (via one of the localisation methods available in the device and the network)
3-10-9 The user can choose the precision of the location that they want to communicate a Street, City or Country for example
3-10-10 If authorised by the Service Provider (GEOLOCATION VALIDITY parameter in section A.1.8.2), the user has the option to enter a validity time for the geolocation information

3.10.3.2 Geolocation PULL

3-10-11 Geolocation information should be made available to any user (notwithstanding whether at home-PLMN or roaming in visiting-PLMN)
3-10-12 Shall be deployed as point to point service between 2 RCS users having the capability
3-10-13 An RCS user (Emitter side) shall have the possibility to retrieve geolocation information from a contact that has Geolocation PULL capability
3-10-14 The service can be accessed through the address book
3-10-15 The service can be accessed also within a call, a Chat or a Group Chat
3-10-16 The contact (Receiving side) shall have the possibility to accept or to deny the request

3-10-17 There is an expiration period for the authorisation granted by the target subscriber. The authorisation is on per application (RCS) and per requesting subscriber basis.
3-10-18 The subscriber shall be able to STOP the authorisation at any time before the expiration period ends
3-10-19 In case of DENY or STOP, the user shall have the possibility to REVOKE the originator of the Geolocation PULL request. In this case, the originator is put in a Geolocation PULL black list
3-10-20 If the Receiving side accepts the demand, geolocation information provided consists of: coordinates (x,y)
3-10-21 If authorised by the Service Provider (GEOLOCATION VALIDITY parameter in section A.1.8.2), the user shall have the option to enter a validity time for the geolocation information when the target user is replying to allow PULL operation

3.10.3.3 Show on a Map

3-10-22 It shall be possible to show the locations of the participants in a 1-to-1 or Group Chat or a call together on a map

3.10.4 Technical Realisation

3.10.4.1 Geolocation PUSH service

The RCS File Transfer service (see previous sections 3.5.4.1 and 3.5.4.3 to 3.5.4.7) is used to convey the geolocation information during a voice or video call (assuming the person the user wants to send his location to is the one in the call).

When there is no such communication context or there is an already established Chat session (1-to-1 Chat or Group Chat), the geolocation information shall be sent directly as a message in a Chat session provided the intended recipient (for a 1-to-1 Chat) or the controlling function (for a Group Chat) supports Geolocation Push. In both cases the same format shall be used which is described in section 3.10.4.3. As for File Transfer via HTTP, message revocation procedures as described in section 3.3.4.1.10, do not apply for 1-to-1 Chat messages carrying geolocation information.

3.10.4.1.1 Geolocation PUSH during a voice or video call

3.10.4.1.1.1 Requester side

The Geolocation PUSH service is proposed to the user if the Service Discovery Process has determined that the target RCS user has the Geolocation PUSH service available. See chapter 2.6 and chapter 2.6.4.1 for Service Discovery. An RCS user having the RCS Geolocation PUSH capability must have also the RCS File Transfer capability

If the user has chosen to provide his/her location through automatic localisation, the RCS user’s device is free to use other locating method(s) rendering the highest possible precision in obtaining geolocation information

The geolocation application interfaces with the RCS File Transfer enabler

Next to the standard SDP parameters for RCS File Transfer, the application of the sender has to set the parameter *type* of the *file-selector* attribute to *application/vnd.gsma.rcspushlocation+xml*

The SIP File Transfer uses a specific IARI that allows routing the primitive to the geolocation application in the target device B. The file transfer name has no meaning in this case.

The file type is *application/vnd.gsma.rcspushlocation+xml*. See the section 3.10.4.3 for more details.

3.10.4.1.1.2 Receiving side

The RCS File Transfer request is routed to the RCS geolocation application (internal routing based on the IARI).

On the receiving side the File Transfer invitation complies with the acceptance rules of RCS File Transfer.

If the transfer is successful, the application triggers the user in a pop up menu to handle the location information.

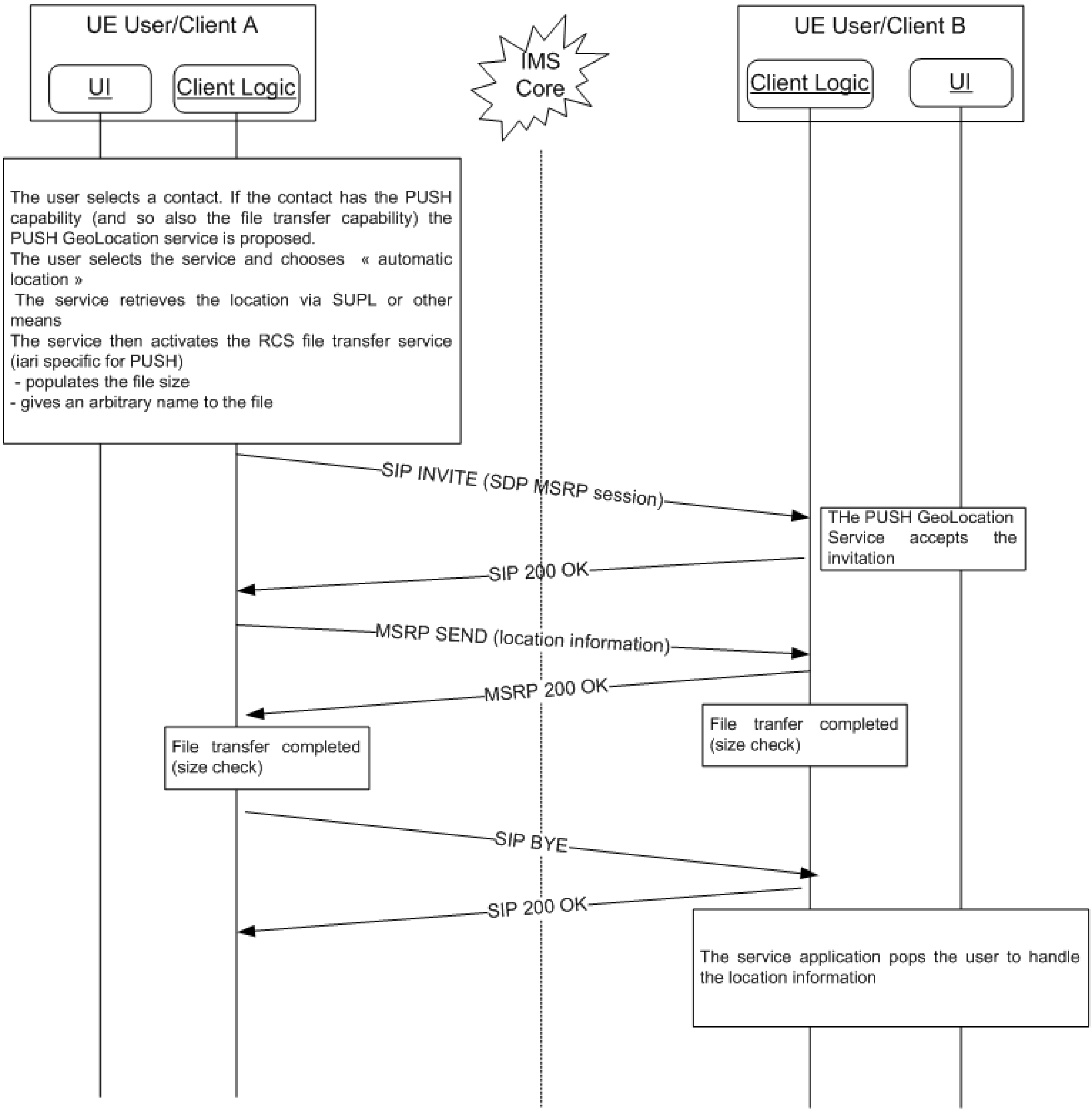

Figure 104: Push of geolocation information during a voice or video call using RCS File Transfer

3.10.4.1.2 Geolocation Push Outside of a voice or video call

When not in a call the Geolocation PUSH service shall send the geolocation information directly in a Chat Message. That allows potentially reusing an already established 1-to-1 or Group Chat session for Geolocation PUSH.

In an active Chat session the sending of the geolocation information shall be possible if

- the Geolocation PUSH content type was included in the a=accept-wrapped-types attribute of the SDP received during the setup of the Chat session and
  - In case of a 1-to-1 session, the contact supports Geolocation PUSH (i.e. the corresponding capability was discovered or was cached)
  - In case of a Group Chat, the Contact header received during the setup of the Group Chat included the Geolocation PUSH IARI tag defined in section 2.6.1.1.2.

When these conditions are fulfilled a client can transfer the geolocation information in a CPIM wrapper that is transferred using an MSRP SEND request with the encapsulated Content-type header of the CPIM message set to *application/vnd.gsma.rcspushlocation+xml*.

3.10.4.1.2.1 1-to-1 Exchange of Geolocation PUSH outside of a voice call

In case a new 1-to-1 session needs to be established when the user wants to transfer geolocation information to a contact that has the Geolocation PUSH capability, the sending client shall generate a SIP INVITE request for a 1-to-1 Chat session as specified in section 3.3.4 and include an additional *Accept-Contact* header field in the SIP INVITE request carrying the Geolocation PUSH IARI defined in Table 11 along with the *require* and *explicit* parameters. This will ensure that the request is routed only to Geolocation PUSH capable devices which shall handle the acceptance of the received SIP INVITE request in the same manner as that of a regular Chat INVITE request (i.e. controlled through the IM SESSION START and IM SESSION AUTO ACCEPT configuration parameters). The Geolocation PUSH XML message body itself (i.e. geolocation information in a CPIM wrapper with the encapsulated Content-type header set to *application/vnd.gsma.rcspushlocation+xml*) shall then be sent as first message in the Chat according to the configured Chat Technology (see section 3.3) based on the configuration parameter CHAT MESSAGING TECHNOLOGY defined in Table 75 in section A.1.4.3.

If there is an active 1-to-1 Chat session with a Geolocation PUSH capable contact, but the *a=accept-wrapped-types* SDP attribute received during the setup of that Chat session did not include the *application/vnd.gsma.rcspushlocation+xml* MIME content type, Geolocation PUSH to that contact will not be available.

3.10.4.1.2.2 Multiparty Exchange of Geolocation PUSH

During Group Chats, the capability to use Geolocation PUSH depends on the controlling function. A Geolocation PUSH capable controlling function shall enable Geolocation PUSH by including the *application/vnd.gsma.rcspushlocation+xml* MIME content type in the *a=accept-wrapped-types* SDP attribute that it provides during the setup of the Group Chat Session and include the Geolocation PUSH IARI tag defined in section 2.6.1.1.2 in the Contact headers that it includes in the SIP INVITE requests and 200 OK responses for the setup of the Group Chat. A Geolocation PUSH capable controlling function shall not distribute Geolocation PUSH information to the participants in the chat that are not Geolocation PUSH capable. A client on which Geolocation PUSH was enabled shall during the setup of the Group Chat indicate to the controlling function that it supports Geolocation PUSH by including the *application/vnd.gsma.rcspushlocation+xml* MIME content type in the a=accept-wrapped-types SDP attribute and the Geolocation PUSH IARI tag defined in section 2.6.1.1.2 in the Contact headers of the SIP INVITE requests and 200 OK responses that it generates. When during a Group Chat the *a=accept-wrapped-types* SDP attribute received by a client or conference focus did not include the *application/vnd.gsma.rcspushlocation+xml* MIME content type or the Geolocation PUSH IARI tag was not provided in the received Contact header, Geolocation PUSH shall not be available for the Group Chat in which the client participates and for a specific client in the Group Chat respectively.

When the users wants to send the Geolocation information to the participants of an existing Group Chat that is idle, a client that is configured to support Geolocation PUSH shall first restart the chat and then send the file in the chat.

When the user wants to send geolocation information to multiple contacts outside of the context of an existing Group Chat, a client that is configured to support Geolocation PUSH shall first start a new Group Chat with the selected contacts and send the Geolocation XML body as first message in the chat.

3.10.4.2 Geolocation PULL service

The Geolocation PULL service is proposed to the user if the capability exchange has determined that the target has the service available and the service operator has authorised the service to be used (see the parameter PROVIDE GEOLOC PULL in Annex A, Table 81).

3.10.4.2.1 Technical solution based on file transfer

The solution is based on an end to end “pull” CPM/SIMPLE IM file transfer.

The format of the file is identical to the format used for the Geolocation PUSH service (see section 3.10.4.3 for the definition of the format).

In a multi-device environment, the file transfer request must be routed to the mobile device. For that purpose, at IMS registration phase, a Broadband device must not register the IARI associated with the Geolocation PULL Service based on File Transfer (see section 2.6.1.1.2) since it cannot offer the service. A broadband device on which the service is enabled may use the service though, but shouldn’t indicate the capability as part of the capability exchange.

3.10.4.2.1.1 Requester side

The Geolocation application builds an OMA SIMPLE IM or OMA CPM (depending on the messaging technology used by the RCS service provider) File Transfer session that includes the SDP attribute a=recvonly in the SIP INVITE request. The requester indicates that he/she wants to receive geolocation information by setting the *type* selector of the *file-selector* attribute (defined in [RFC5547]) to *application/vnd.gsma.rcspushlocation+xml*.

The inclusion of the dedicated IARI value for Geolocation PULL (defined in Table 11) in the *Contact* header field and along with *require* and *explicit* tags in a dedicated *Accept-Contact* header field allows identifying the request as a Geolocation PULL request at the receiver side.

The behaviour of the application depends then on the response to this SIP INVITE request for File Transfer:

- A SIP 200 OK Response: The operation is authorised by the target user. The MSRP session associated with the File Transfer exchange allows the application to retrieve the location information
- 603 Decline: The operation is not authorised by the target user. Reason can be:
    - The target user has explicitly not authorised the operation
    - The requester is blocked by the target user

  In this case, the requester is not allowed to send a similar Location Pull operation to this contact for a duration that is controlled by the service provider (GEOLOCATION PULL BLOCK TIMER defined in Annex A in Table 81)

- A Time out : the operation is not successful, there is no restriction on the application for resending a similar request towards the target user

3.10.4.2.1.2 Receiving side

On receiving a Geolocation PULL in a File Transfer request, the following steps are processed:

- The Geolocation PULL request has to be explicitly authorised by the receiver.
- When authorizing the request the user must provide the following:
    - The accuracy of the location they want to provide to this requester.
    - Optionally a validity time: This is the time the user estimates his current location to remain stable. The value is also controlled by the Service Provider through the GEOLOCATION VALIDITY parameter (see Table 81 in section A.1.8.2)
    - Optionally an authorisation validity time (during this time, any other requests from the sender are automatically accepted by the application without consulting the end user for authorisation). When not provided, by default the authorisation is granted for one request only. The authorisation timer is a client internal timer that is not visible on the UNI interface

- If the request is authorised (either explicitly by the user or automatically when the authorisation validity time has not elapsed), the Geolocation PULL application fetches the location, establishes the File Transfer Session by returning a 200 OK response and uses that to return a file indicating the user’s location with indicated accuracy
- If the authorisation is denied by the user (either because the requesting user is in the local black list, or the user explicitly rejects the Geolocation PULL request), a SIP *603 Decline* response is returned to the requester

- If the authorisation is pending (i.e. the user has not answered the authorisation request before the SIP INVITE transaction timer has elapsed), no response is sent back. This will result in a timeout of the SIP INVITE transaction at the requester's side

3.10.4.2.1.3 Flows

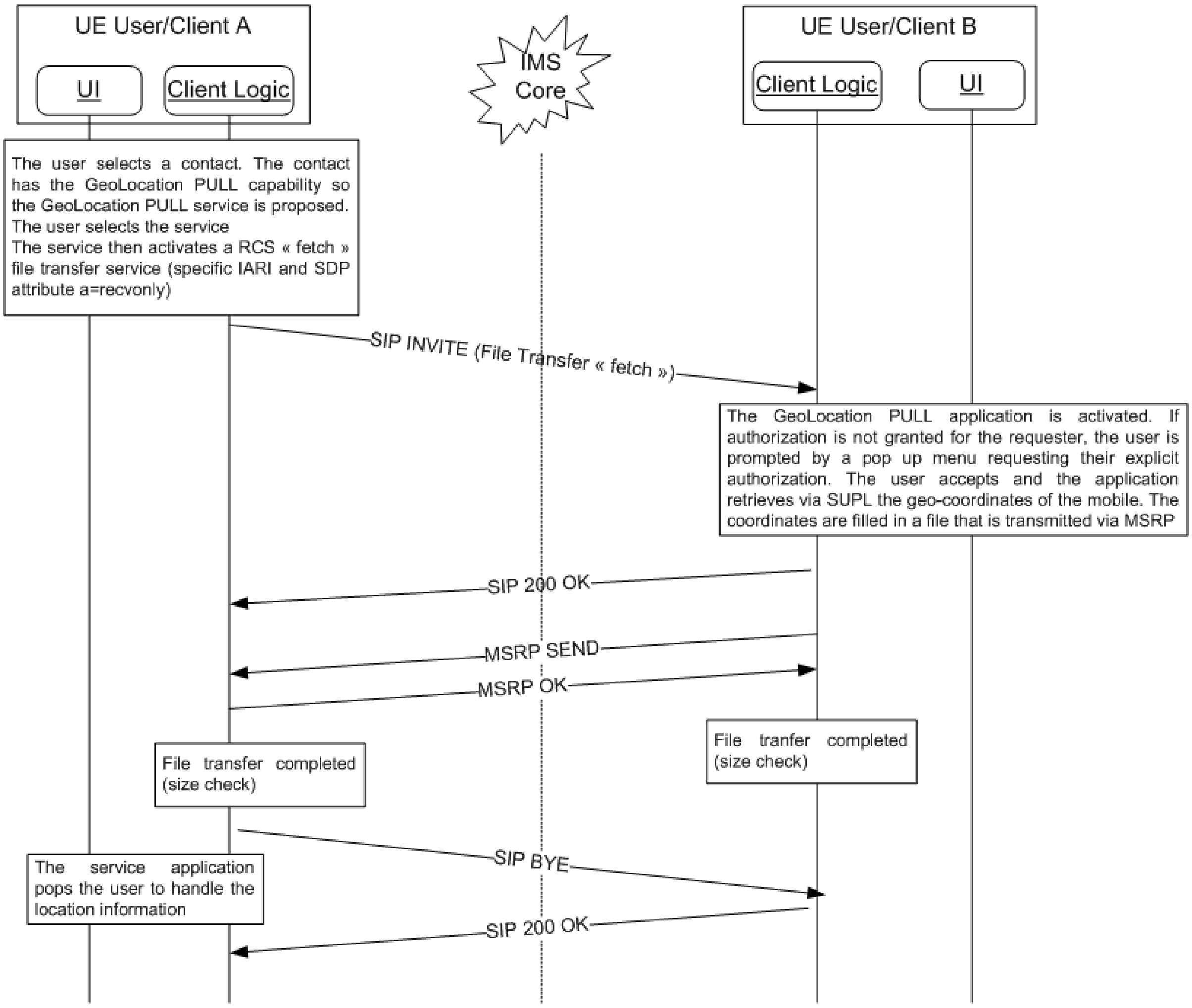

Figure 105: Pull of geolocation information: Success case

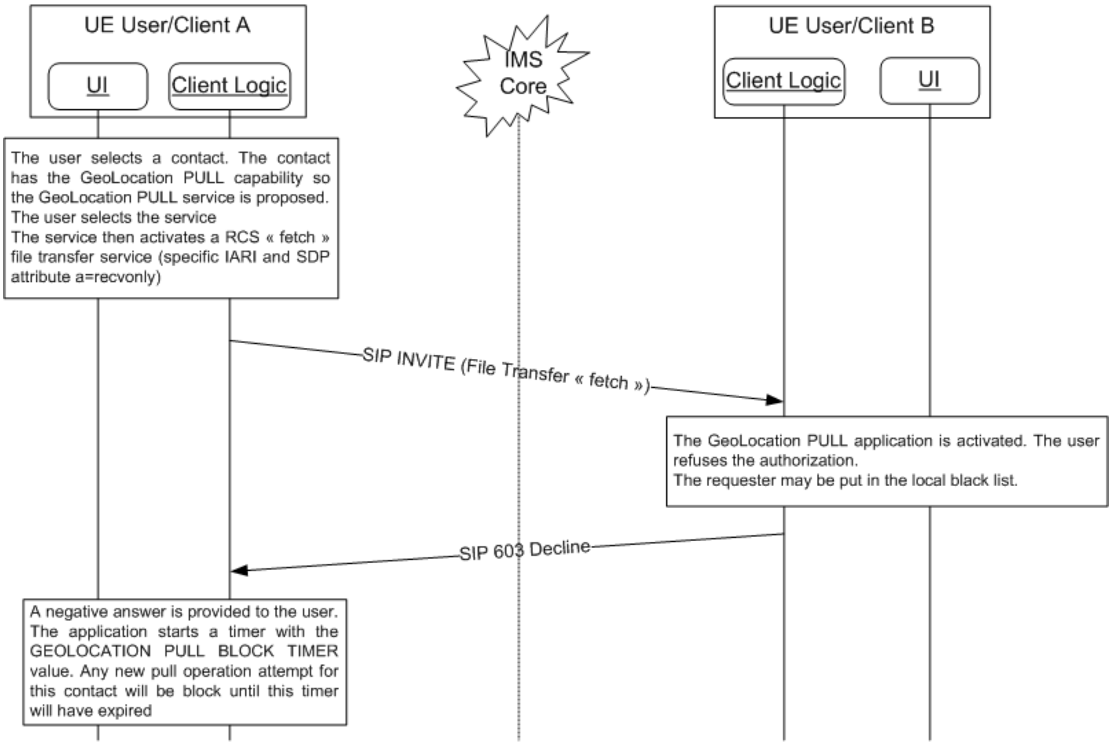

Figure 106: Pull of geolocation information: The target user refuses to give their authorisation for the operation

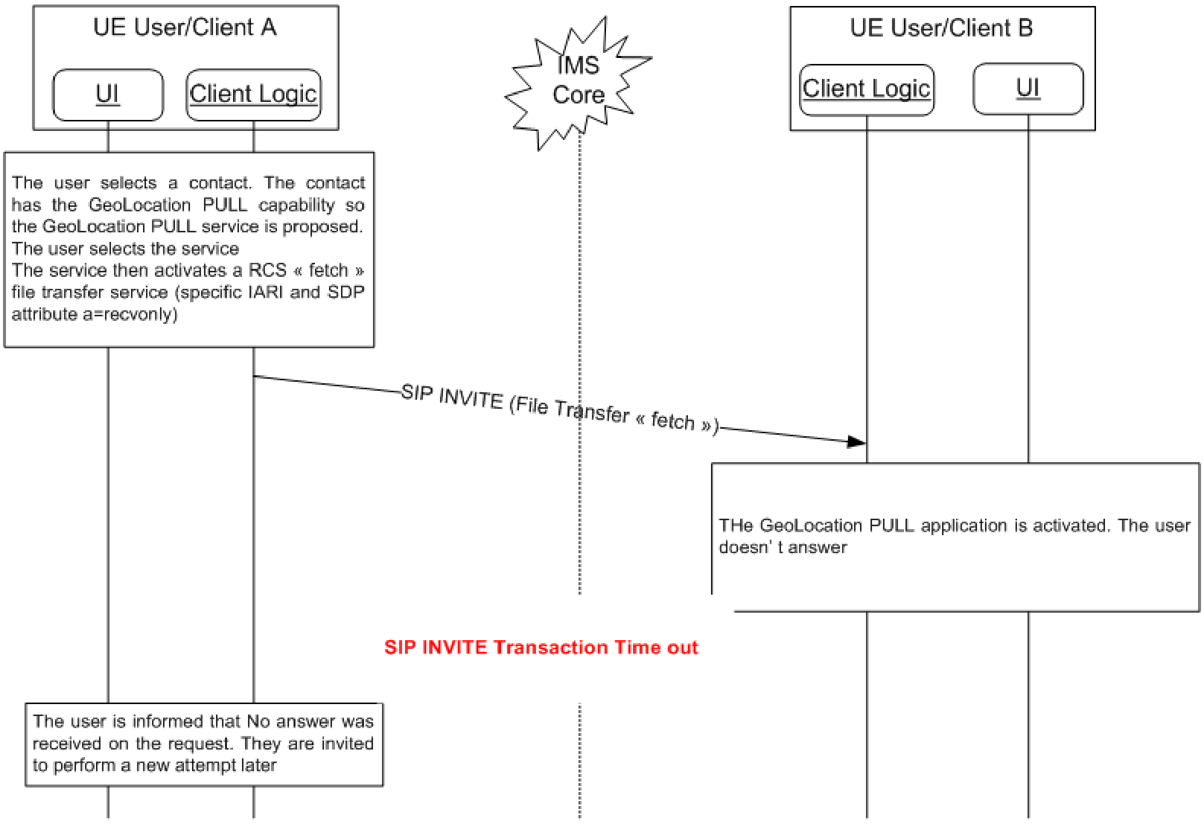

Figure 107: Pull of geolocation information using File Transfer: The target user doesn’t answer

3.10.4.3 Location Information format

3.10.4.3.1 General

The format of the information re-uses the general structure of the RCS XML Presence data. It uses a subset of RCS SPI data definition adapted to RCS Location information

The following XML schema is defined:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:gsma:params:xml:ns:rcs:rcs:geolocation"
            xmlns:xs="http://www.w3.org/2001/XMLSchema"
            xmlns="urn:gsma:params:xml:ns:rcs:rcs:geolocation"
            elementFormDefault="qualified"
            attributeFormDefault="unqualified">
    <xs:element name="rcsenvelope">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="rcspushlocation">
                    <xs:complexType>
                        <xs:sequence>
                            <xs:any namespace="##other" processContents="lax"
                            minOccurs="0" maxOccurs="unbounded"/>
                            <xs:element name="timestamp">
                                <xs:simpleType>
                                    <xs:restriction base="xs:dateTime"/>
                                </xs:simpleType>
                            </xs:element>
                        </xs:sequence>
                        <xs:attribute name="id" type="xs:ID" use="required"/>
                        <xs:attribute name="label" type="xs:string" use="optional"/>
                    </xs:complexType>
                </xs:element>
                <xs:any namespace="##other" processContents="lax" minOccurs="0"
                maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="entity" type="xs:anyURI" use="required"/>
        </xs:complexType>
    </xs:element>
</xs:schema>
```

Table 52: Geolocation PUSH Envelope XML schema

3.10.4.3.2 RCSPushLocation data model

Attribute	Specification	Comment
Person: <rcsenvelope> -> <rcspushlocation>	Table 52	Each client only publishes one <rcsenvelope> and one <rcspushlocation> element. The rcspushlocation element may have a label that can be used to tag the nature of the location (e.g. indicate that it’s the home or provide an address, name of restaurant, etc.). If no label is provided, the location that is shared is assumed to be the sharing user’s own position.

Attribute	Specification	Comment
Time Zone <rcspushlocation> -> <time-offset>	Table 52, [RFC4480] and [Presence2.0_DDS]	The geolocation application may use this element to provide information on the current time zone See following chapter section for more information on the handling of the expiry of this information
Geographical Information <rcspushlocation> -> <geopriv> -> <location-info> -> <usage-rules>	Table 52, [RFC5491] and [Presence2.0_DDS]	This element can be used to provide geographical location information. The accuracy of which can be controlled by the user. See following section for more details on its encoding and on the handling of the expiry of this information
Timestamp: <rcspushlocation> -> <timestamp>	Table 52, [RFC4479]	Timestamp when the location information was pushed

Table 53: RCSPushLocation data model attributes

3.10.4.3.3 RCSLocation information

RCS clients shall not include a “*from*” attribute in the <*time-offset*> element. RCS clients shall ignore it when received.

RCS clients can provide (if authorised by the Service Provider) an "*until*" attribute in that element. The user will populate the validity time of the information with a value that will not exceed a data configuration value (see section A.1.8.2).

NOTE1: this behaviour deviates from SPI where this element is mandatory.

RCS clients shall not include the optional description attribute in the <*time-offset*> element as this overlaps with the Location Type. RCS clients shall ignore it when received.

The geographical information will be provided as geographic coordinates. As specified for the “Geographical Location” building block in [Presence2.0_DDS], encoding will use the <*geopriv*>→<*location-info*> and <*geopriv*>→<*usage-rules*> elements.

The optional <*usage-rules*> element shall contain, if present, only a "*retention-expiry*" element. The RCS client shall set the "*retention-expiry*" to the same value as the “*until*” attribute mentioned above.

NOTE2: this behaviour deviates from SPI where this element is mandatory

The <*location-info*> published by an RCS Geolocation client will contain geographical information using the GML 3.1.1 Feature Schema (see [GML3.1.1]) which is the mandatory format to be used in the <*location-info*> element. The civic location format shall not be used by RCS and location information encoded in that way will be ignored by RCS clients when received.

RCS client will within the <*location-info*> element represent an exact position by providing a GML <point> element and an inaccurate position as a <*circle*> element, both referring to the EPSG::4326 spatial reference schema as described in [RFC5491].

The coordinates of either the centre of this circle or the exact position will be represented with a single GML *<pos>* element with the actual coordinates as value.

The radius of the circle will be represented in meters, which will be indicated by setting the unit of measure attribute of the radius element to the value of EPSG::9001 as described in [RFC5491].

The text value (that is, the *<place-type>* element) shall not exceed a Service Provider configured value (see section A.1.8.2).

In case of Geolocation PUSH, the text is entered by the user.

In case of Geolocation PULL, a text can be entered automatically by the application (for example, the application, depending on the location accuracy allowed by the user, can fill a text that gives information on the user's geographical position such as street, number and city name if a high accuracy position is allowed, or only a city name if the user only allows to provide a less precise location).

An RCS client shall ignore any other type of data provided in the *<location-info>* element.

The EPSG format requires that the coordinate representation is defined by the coordinate supplier. RCS client will always provide the coordinates in WGS 84 (latitude, longitude) decimal notion as described in [RFC5491], providing the latitude and longitude as "double"-encoded decimal numbers (as specified in [GML3.1.1]) representing the degrees, separated by a space starting with the latitude. Negative values represent Southern and Western hemisphere respectively.

The following gives an example of RCS Location information data:


```
<?xml version="1.0" encoding="UTF-8"?>
<rcsenvelope xmlns="urn:gsma:params:xml:ns:rcs:rcs:geolocation"
        xmlns:rpid="urn:ietf:params:xml:ns:pidf:rpid"
        xmlns:gp="urn:ietf:params:xml:ns:pidf:geopriv10"
        xmlns:gml="http://www.opengis.net/gml"
        xmlns:gs="http://www.opengis.net/pidflo/1.0"
        entity="tel:+1234578901">
        <rcspushlocation id="a1233" label="meeting location">
                <rpid:time-offset rpid:until="2012-03-15T21:00:00-05:00">-300</rpid:time-offset>
                <gp:geopriv>
                        <gp:location-info>
                                <gs:Circle srsName="urn:ogc:def:crs:EPSG::4326">
                                        <gml:pos>26.1181289 -80.1283921</gml:pos>
                                        <gs:radius uom="urn:ogc:def:uom:EPSG::9001">10</gs:radius>
                                </gs:Circle>
                        </gp:location-info>
                        <gp:usage-rules>
                                <gp:retention-expiry>2012-03-15T21:00:00-05:00</gp:retention-expiry>
                        </gp:usage-rules>
                </gp:geopriv>
                <timestamp>2012-03-15T16:09:44-05:00</timestamp>
        </rcspushlocation>
</rcsenvelope>
```


Table 54: Example of location information data

3.10.5 NNI and IOT considerations

The NNI interfaces for geolocation services shall behave according to the procedures described in section 2.12 and the documents it refers to.

3.10.6 Implementation guidelines and examples

3.10.6.1 Geolocation PUSH

The Geolocation PUSH feature can be selected by an RCS user whenever it makes sense to share her/his location information with other RCS users, i.e.:

- From the general “share menu” or
- Inside a call / video call
- Or inside a Chat or a Group Chat

On the receiver side, a “pop up” menu advertises the user that an RCS user is communicating some location information

NOTE: For locations carrying a label, the client may offer the user to permanently store this location on the device.

3.10.6.1.1 Show in a Map

When during a call or 1-to-1 chat, a Geolocation Push is received from the conversation partner providing their location (i.e. not carrying a label as described in section 3.10.4.3.2), the client should request the user whether they want to share their location unless it was shared recently with that contact (i.e. the validity time should not have expired). If the user accepts or a valid location was already provided, the client should show both locations on a map and offer the user the option to refresh his location.

NOTE: Also other locations (i.e. carrying a label) shared during the conversation may be shown on the map.

When a contact sends a location carrying the same label or carrying no label and previously a different location with the same label (or no label) was shared by that same contact, the previous location should be removed from the map and the new location should be shown instead.

Similarly if in a Group Chat the location from another participant is received, the user should be requested whether to share his location with the other participants unless that was done recently already. All received locations of other participants should be shown on a map including their own location if it was shared, with the possibility to share a refresh or initial version of the user’s own location with the other participants.

This leads to following UX:

- For the initial sender

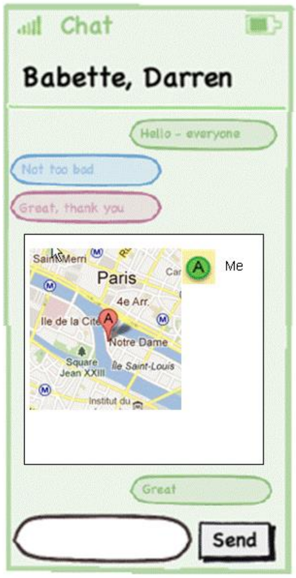

Figure 108: Show on Map: Initiator

- For the initial recipient(s) that have not shared their location yet

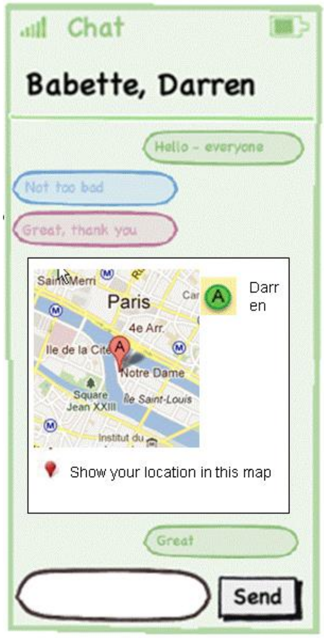

Figure 109: Show on Map: initial recipients

- For a group chat participant once the user has shared their location

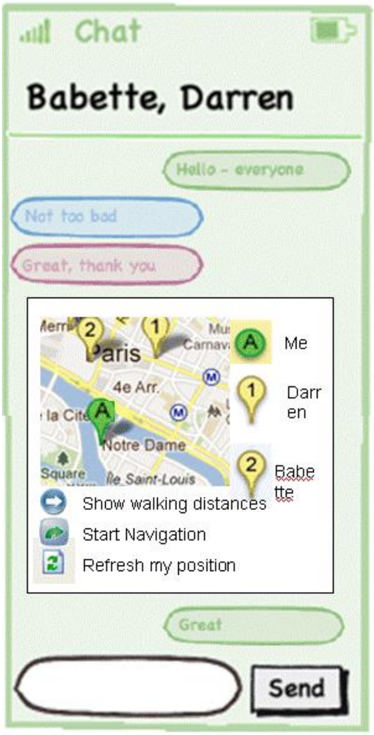

Figure 110: Show on Map: Group Chat participant after sharing own location

3.10.6.2 Geolocation PULL

The Geolocation PULL feature can be selected by an RCS user in same circumstance as the Geolocation PUSH feature, i.e.:

- From the general “share menu” or
- Inside a call / video call
- Or inside a chat or a Group Chat

At the receiver side:

- A pop up menu is presented by the application when a Geolocation PULL request is received and no automatic authorisation is granted to the requester (either because this is the first request received from this user or because the authorisation validity time of a previously authorised request has expired.)
- The user then has the possibility to accept or deny sharing their location with the requester
- If the user accepts, they have:
  - Depending on Service Provider policy (see GEOLOCATION VALIDITY parameter defined in section A.1.8) the option to associate a validity time for the information.
  - The possibility to define the duration the authorisation is granted to the Requester
  - The option to choose the level of accuracy for the location that will be provided to this requester (for example, Country, City, Street)
- At any time, in the address book, the user can activate a menu to revoke their authorisation for Geolocation PULL by a dedicated contact.

3.11 Audio Messaging

3.11.1 Feature description

This feature enables an RCS user to record and/or send an audio message to his RCS contacts. An RCS Recorded Audio Message (RRAM) can be sent to one or more contacts. When the RRAM is recognized by the receiving RCS Client as being an audio message, it is handled consequently (as described further in this feature's section).

3.11.2 Interaction with other RCS features

The Audio Message feature is linked to the File Transfer service that conveys the RRAM to the recipient.

3.11.3 High Level Requirements

3-11-1 A RRAM can be sent to one or more contacts.
3-11-2 The message display will show the time, date and duration of each message.
3-11-3 Message recording shall be limited to a maximum duration

3.11.4 Technical Realization

3.11.4.1 RCS Recorded Audio Message format

An RCS client shall encode the audio message using the Adaptive Multi-Rate (AMR) codec.

The RRAM shall be formatted in the file format defined in [RFC4867].

The transport of RRAM uses the File Transfer features:

- standard notification mechanism
- store and forward when available
- auto-acceptance rules for File Transfer
- Technology is either HTTP or MSRP based on the FT DEFAULT MECH configuration parameter (see Table 76) and the supported File Transfer technologies according to the capability exchange.

3.11.4.2 Sender procedures

3.11.4.2.1 Recording

When the Audio Message is selected via the User Interface, the Client shall record an audio file via the device's microphone.

The duration of the RRAM shall be limited to a maximum duration of 10 minutes. The Client shall automatically stop the recording when this limit is reached.

Once recorded, the content should automatically be packaged into the file format described in section 3.11.4.1.

3.11.4.2.2 Sending

When sending a RRAM to a contact, the RRAM is transported via the available File Transfer service (see section 3.5) taking into account the supported technology (i.e. MSRP or HTTP). The File Disposition shall be set to '*render*'.

NOTE: '*render*' means that the content of the file can be played directly from the Chat application upon user action.

When using transport over HTTP, in complement to the procedures of section 3.5.4.8.3.1, the Client shall put the length of the RRAM in the playing-length element of the File transfer via HTTP message body content, as defined in Table 55.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:gsma:params:xml:ns:rcs:rcs:rram"
          xmlns:xs="http://www.w3.org/2001/XMLSchema"
          xmlns="urn:gsma:params:xml:ns:rcs:rcs:rram"
          elementFormDefault="qualified"
          attributeFormDefault="unqualified">
     <xs:element name="playing-length">
          <xs:simpleType>
               <xs:restriction base="xs:integer"/>
          </xs:simpleType>
     </xs:element>
</xs:schema>
```

Table 55: Extension to File Transfer via HTTP message body schema for Audio Message

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<file xmlns="urn:gsma:params:xml:ns:rcs:rcs:fthttp"
   xmlns:am=" urn:gsma:params:xml:ns:rcs:rcs:rram“>
     <file-info type="file" file-disposition="[file-disposition]">
          <file-size>[file size in bytes]</file-size>
          <file-name>[original file name]</file-name>
          <content-type>[MIME-type for file]</content-type>
          <am:playing-length>[duration of the rram]</am:playing-length>
          <data url ="[HTTP URL for the file]" until="[validity of the file]"/>
     </file-info>
<file>
```

Table 56: Example of Audio Message Transfer using File Transfer via HTTP

3.11.4.3 Receiver procedures

On the receiving side, when a File Transfer request is received with the file-disposition set to "render" and the content is recognized as corresponding to the file format described in section 3.11.4.1, rather than announcing the transfer as a File Transfer, the UI shall announced that an audio message is received. If accepted or auto-accepted, the received content shall be displayed in the corresponding 1-to-1 or Group Chat thread as an audio message with the option to play it. The RRAM shall not be played automatically. The Display Notification (if requested) shall be sent when the playing of the file is started.

3.11.5 NNI and IOT considerations

No specific guidelines apply other than what is already defined in Section 2.12

3.11.6 Implementation guidelines and examples

From the UX point of view, two possible entry points to the Audio Message service are:

1. <u>Address book/Call-log</u>: An audio message can be initiated with any contact.

2. Chat window: From the Chat (one-to-one Chat only) window an audio message can be initiated using the relevant menu item. The experience is identical to the address book/call-log.

Audio messages can be shared easily by all RCS users within 1-to-1 and group chat sessions by simply holding down a soft key/button to record the message. This shall also be possible via an entry point on the contact card.

Audio messages are received within the chat or group chat thread associated with the contact that has sent the message.

3.12 Extension

3.12.1 Feature description

This feature enables an Extension to use the RCS infrastructure to communicate with other RCS entities.

3.12.2 Interaction with other RCS features

Due to its nature, the Extension feature interacts with any other RCS feature; e.g. invoke a feature.

3.12.3 High Level Requirements

3-12-1 An Extension shall be uniquely identified.
3-12-2 An Extension may use RCS features (e.g. File Transfer) when communicating with other RCS entities.
3-12-3 Some Extensions may require to use RCS features (e.g. Chat) only between the same instances of those Extensions.
3-12-4 An Extension may generate its own specific traffic.
3-12-5 The Extension specific traffic may be message based.
3-12-6 The Extension specific traffic may be real time based.
3-12-7 Any traffic generated by an Extension shall be identified in the network as being issued from this Extension.
3-12-8 A Service Provider shall be able to revoke an Extension.

3.12.4 Technical Realisation

3.12.4.1 Communication from Extension not specifically targeted towards another specific Extension

An Extension is allowed to use any session based RCS feature just like any RCS client entity (e.g. Video Share). In this case, when initiating a session, the standard procedures defined in the section of this document corresponding to the feature to use apply with the following modification:

When initiating a session, the SIP INVITE request initiated by the Extension shall include the Extension’s IARI tag in the Contact header.

When accepting a session, the SIP 200 OK sent by the Extension shall include the Extension’s IARI tag in the Contact header.

3.12.4.2 Communication between specific Extensions

This kind of communication is established only between instances of the same Extensions (i.e. they have the same IARI tag) and is only possible when the ALLOW RCS EXTENSIONS parameter (as defined in section A.1.14) is set to 1 on the Client.

As a general rule for the following sub-sections, when setting a session, the SIP INVITE requested by the Extension shall include:

- The service feature tag (e.g. an ICSI when such ICSI is defined) in the Accept-Contact header,
- The Extension's IARI tag with the *require* and *explicit* parameters in a dedicated Accept-Contact header, provided that the Service itself is not defined by an IARI (e.g. content sharing).

NOTE: Services defined by IARIs cannot be part of those communications between specific Extensions.

3.12.4.2.1 Communication derived from RCS services

3.12.4.2.1.1 Messaging based channel

This communication type uses the messaging ICSIs and is processed by the Messaging Server. It makes use of the standard Messaging Server processing (e.g. store and forward) with the following differences.

If the incoming SIP request contains the *require* and *explicit* parameters on the Accept-Contact header containing an IARI tag:

- The push of stored data shall be done only towards a Client hosting the same IARI as the one that has generated the stored data.
  - When storing data due to temporary unavailability of the intended recipient, the Messaging Server shall store the associated IARI tag pertaining to the IARI producing the data, including the require and explicit parameters.
  - Before pushing stored data to a reconnecting RCS Client, the Messaging Server shall check if the associated stored IARI tag is used by the reconnecting Client (e.g. via information provided by the third party register).
- No automatic storage is done in the Common Message Store.
- There is no interworking with SMS/MMS.

NOTE: The above functionality is required to be brought into the relevant OMA specifications.

3.12.4.2.1.2 Real time based channel

No specific processing is required.

3.12.4.2.2 RCS Extension to Extension Service

3.12.4.2.2.1 Service definition

An RCS user's instance of an Extension can communicate with another RCS user's instance of an Extension (Extension to Extension) using their specific data. This data specific to the Extension is not covered in this specification. However, this specification defines a new service aiming at transporting this kind of data: the Extension to Extension service.

The Extension to Extension ICSI is defined as shown in Table 57:

Extension to Extension ICSI	Tag
Value carried in an Accept-Contact or Contact header	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.gsma.rcs.extension"
Value carried in a P-Preferred-Service or P-Asserted-Service header	urn:urn-7:3gpp-service.ims.icsi.gsma.rcs.extension

Table 57 : Extension to Extension ICSI values

3.12.4.2.2.2 Initiating an RCS Extension to Extension Session

Handling at Initiating Nodes

The RCS Client SHALL send an initial SIP INVITE request according to the rules and procedures of [3GPP TS 24.229]. In this SIP INVITE request, the RCS Client:

1. shall include the address of the target RCS contact in the Request-URI;
2. shall include an Accept-Contact header field with the Extension to Extension ICSI 'urn%3Aurn-7%3A3gpp-service.ims.icsi.gsma.rcs.extension', percent encoded as per [3GPP TS 24.229] section 7.2A.8.2 "*Coding of the ICSI*" in a g.3gpp.icsi-ref media feature tag with the *require* and *explicit* parameters;

NOTE: This step covers the first bullet of section 3.12.4.2.

3. shall include an Accept-Contact header field with the Extension's IARI tag with the *require* and *explicit* parameters;

NOTE: This step covers the second bullet of section 3.12.4.2.

4. shall set the P-Preferred-Service header field with the value of the Extension to Extension ICSI 'urn:urn-7:3gpp-service.ims.icsi.gsma.rcs.extension';
5. shall include a Contact header field with the Extension to Extension ICSI '*urn%3Aurn-7%3A3gpp-service.ims.icsi.gsma.rcs.extension*' percent encoded as per [3GPP TS 24.229] section 7.2A.8.2 "*Coding of the ICSI*" in a *+g.3gpp.icsi-ref* media feature tag, and with the Extension's IARI tag;
6. shall include the address of the originating RCS Client that has been authenticated as per section 2.5 and [3GPP TS 24.229];
7. shall include a User-Agent header field as specified in Appendix C.5 "Extension to Extension ICSI Release Version in User-Agent and Server headers";

8. should include a Session-Expires header field with the refresher parameter set to "uac" according to the rules and procedures of [RFC4028];
9. shall include a MIME SDP body as an SDP offer as described in section 3.12.4.2.2.4;
10. shall send the SIP INVITE request according to the rules and procedures of [3GPP TS 24.229].

On receipt of the SIP 200 "OK" response to the initial SIP INVITE request the RCS Client shall handle the response according to the rules and procedures of [3GPP TS 24.229], with the following clarifications:

1. The RCS Client shall start a SIP session timer using the value received in the Session-Expires header field according to the rules and procedures of [RFC4028].
2. The RCS Client shall generate and send a SIP ACK request as an acknowledgement of the final response according to the rules and procedures of [RFC3261].
3. The RCS Client shall establish the Media Plane as per [3GPP TS 24.229].

On receipt of a SIP error response to the initial SIP INVITE request the RCS Client SHALL handle the response according to the rules and procedures of [3GPP TS 24.229], with the following clarifications:

1. The RCS Client may indicate to the user that the session could not be established;
2. The RCS Client shall generate and send a SIP ACK request as an acknowledgement of the final response according to the rules and procedures of [RFC3261].

Handling at Intermediate Nodes

Intermediate nodes (e.g. access gateways, application servers) may stay in the media path depending on Service Provider policy.

Handling at Terminating Nodes

On receipt of the SIP INVITE request the RCS Client shall check if the Extension as indicated by the IARI in the Accept-Contact header is running on the device:

1. If not, the RCS Client SHALL respond with a SIP 403 Forbidden error with a Warning header set to "*Unsupported Extension*".
2. If yes, the RCS Client
    a) shall respond with a SIP 200 OK, with a valid SDP offer as per section 3.12.4.2.2.4 if the session is accepted and start a SIP session timer and take on the role of "uas" according to the rules and procedure of [RFC4028], and establish the Media Plane as per [3GPP TS 24.229], or
    b) shall respond with a SIP 603 Decline if the session is not accepted.

If the Client is already involved in an Extension to Extension session with the same contact and the same Extension (i.e. same IARI), it shall terminate the ongoing session as per section 3.12.4.2.2.3 before accepting the new one.

In a multi-device scenario, if more than one RCS Client receives the incoming SIP INVITE request because of forking by the IMS core, as per [RFC3261], only the RCS Client that responds first will remain in the session; the other session will be torn down by the IMS core. If a client responds with SIP 603 Decline, no session at all is set up as per [RFC3261]. If no RCS Client responds, the request will be timed-out.

3.12.4.2.2.3 Terminating an RCS Extension to Extension Session

To close an Extension to Extension session due to an explicit closing request from the Extension, the Client:

1. shall generate a SIP BYE request according to the rules and procedures of [3GPP TS 24.229], with the Reason Header field as defined in [RFC3326] with the protocol-value set to SIP, the protocol-cause set to *200* (e.g., *SIP;cause=200*);
2. shall send the SIP BYE request according to the rules and procedures of [3GPP TS 24.229];
3. shall release all Media Plane resources corresponding to the Extension to Extension session being closed.

A client shall close an Extension to Extension session when it has been idle for longer than the value configured for the IM SESSION TIMER configuration parameter defined in section A.1.4.3. In this case, the Client:

1. shall generate a SIP BYE request according to the rules and procedures of [3GPP TS 24.229], with the Reason Header field as defined in [RFC3326] has a protocol-value set to SIP and a protocol-cause set to *200*;
2. shall send the SIP BYE request according to the rules and procedures of [3GPP TS 24.229];
3. shall release all Media Plane resources corresponding to the Extension to Extension session being closed when a final response to that BYE request is received.

When receiving a SIP BYE request, the client shall

1. shall generate a SIP 200 OK response according to the rules and procedures of [3GPP TS 24.229];
2. shall send the SIP 200 OK response according to the rules and procedures of [3GPP TS 24.229];
3. shall release all Media Plane resources corresponding to the Extension to Extension session being closed.

NOTE: When the Extension wants to send further traffic to the other client after the session has been closed, a new session shall be started as described in section 3.12.4.2.2.2.

3.12.4.2.2.4 SDP Contents

SDP Contents when Initiating a Session

An initiating entity (e.g. an RCS Client) SHALL populate the SDP of an Extension to Extension session invitation request to match the Media Streams that are requested by the pertaining Extension. Therefore the initiating entity shall include in the SIP INVITE request a MIME SDP body as an SDP offer according to the rules and procedures of [3GPP TS 24.229]. The SDP offer shall contain media descriptions matching the requested Media Streams according to the following clarifications:

- When including an offer for a Media Stream using MSRP, the initiating entity shall include a media description according to the rules and procedures of [RFC4975] with the *a=max-size* parameter set to the RCS configuration parameter EXTENSIONS MAX MSRP SIZE (see Annex A.1.14), and *a=accept-types* shall only include the *text/plain* MIME type. Also MSRP Failure Reports shall be requested and MSRP Success Reports SHALL NOT be requested.
- When including an offer for a Media Stream for real-time continuous Media, using RTP/RTCP, the initiating entity shall include a media description according to [RFC3550], [3GPP TS 24.229], [3GPP TS 26.114] and [3GPP TS 26.141], and make use of pre-conditions.

NOTE: For RTP there may be a need to control QoS. How this is done from the network is not covered here. Therefore Extensions making use of RTP can only be provided by Extensions known by the operator.

SDP Handling at Intermediate Nodes

Intermediate nodes shall include the contents of the SDP they received in the SDP they send out, in accordance with the rules and procedures of [3GPP TS 24.229] and [RFC3264]. Specific attributes in the SDP may be modified for the following reasons:

- To modify IP-address and port information to insert the intermediate entity in the media path of the session.

All modifications shall be done according to the rules and procedures of [RFC3264] and the respective Media Stream standards (i.e. [RFC4975] for MSRP-based media description and [RFC3264] and [RFC3550] for RCP/RTCP-based media descriptions).

SDP Handling at Terminating Nodes

A terminating entity (e.g. an RCS Client) shall process an incoming SDP and accept, modify or reject the Media Streams requested in the incoming SDP as defined by [3GPP TS 24.229] and [RFC3264]. The terminating entity SHALL handle the media descriptions according to the following clarifications:

- Media descriptions for a Media Stream for messages, using MSRP, shall be handled and responded to according to the rules and procedures of [RFC4975], with the a=max-size parameter set to the configuration parameter EXTENSIONS MAX MSRP SIZE (see Annex A.1.14), and *a=accept-types* shall only include the *text/plain* MIME type. Also MSRP Failure Reports SHALL be requested and MSRP Success Reports shall not be requested.
- Media descriptions for a Media Stream for real-time continuous Media, using RTP/RTCP, shall be handled and responded to according to the rules and procedures of [RFC3550], [3GPP TS 24.229], [3GPP TS 26.114] and [3GPP TS 26.141] and make use of pre-conditions.

NOTE: For RTP there may be a need to control QoS. How this is done from the network is not covered here. Therefore Extensions making use of RTP can only be provided by Extensions known by the operator.

3.12.4.2.2.5 MSRP Session Handling

Clients in a session set up for MSRP shall respect the value of the a=max-size parameter from the MSRP SDP which is set using the configuration parameter EXTENSIONS MAX MSRP SIZE (see Annex A.1.14), to limit the size of content sent within the session. Also, only text/plain MIME types shall be transferred. If larger messages or files or other MIME types are to be transferred, the RCS File Transfer feature shall be used using procedures from section 3.12.4.2.1.1.

When no response is received to an MSRP SEND, the rules and procedures of [RFC4975] are followed with the following clarification:

- The client not receiving an MSRP SEND response should set the *cause=503* along with an optional protocol-text (e.g. *SIP;cause=503;text="Service Unavailable"*) in the SIP BYE request it generates. The client should indicate to the user that an error occurred when sending the message in the MSRP SEND.

3.12.4.3 Extension revocation

A Service Provider shall be able to control the use of an Extension by an RCS Client.

An Extension control request can be triggered by the network by sending a EUCR system request with type *urn:gsma:rcs:extension:control* as specified in section 2.10.

When the Client receives such a request, it shall take the following actions:

1. Reply to the request with a 200 OK response
2. For each (<IARI>,<duration>) pair contained in the Data attribute, block the Extension matching the <IARI> from accessing the RCS infrastructure for <duration> seconds.
   A duration of ‘0’ means that the corresponding Extension shall no longer access the RCS infrastructure.
   No action is required from the Client in case a IARI is not locally matched with an Extension.

A new request received for an IARI already processed in a previous request shall override that previous request. An Extension which has been blocked (i.e. duration 0) can thus for example be unblocked (e.g. duration 10) via a new EUCR.

3.12.5 NNI and IOT considerations

No specific guidelines apply other than what is already defined in Section 2.12.

3.12.6 Implementation guidelines and examples

From the UX point of view, many possibilities are offered via Extension. These are out of scope of this specification.

3.13 RCS VV-Mail

3.13.1 Feature description

This feature enables the RCS user to access and manage the voicemail messages from the RCS client. The RCS user can perform the following voicemail messaging features from the RCS client:

- Access voicemail messages from several entry points like common VVM mailbox, native messaging thread or call log.
- Listen to the voicemail messages or delete the voicemail messages.
- Synchronize voicemail message status among users multiple devices including when the messages are read and deleted using the Telephony User Interface (TUI).

The voicemail messages are stored in the CPM Common Message Store under the voicemail application specific folders as described in [RCS-CPM-MSGSTOR-ENDORS].

3.13.2 Interaction with other RCS features

When RCS VV-Mail is deployed, the Standalone Messaging service is used for new message notification to RCS clients following rules shall apply for messages that include the RCS VV-Mail feature tag *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.vvm"*:

- The Standalone deferred messaging logic applies to the voicemail notification event.
- The message shall not be stored in the Common Message Store.
- There is no change to the support of a user's multiple devices and a message shall be delivered to all clients of a user that are online and that are capable of handling the RCS VV-Mail Standalone new message notification.
- The interworking shall be performed as defined in section 3.13.4.6.

3.13.3 High Level Requirements

The following list of high level requirements applies to rcsVVM:

- Clients/devices:

3-13-1 Update voice mail message status (e.g. read, delete or archive)
3-13-2 Send voice mail status update notifications to other devices

- Messaging Server: In addition to the requirements in other sections

3-13-3 Interwork voice mail status notifications to SMS/MMS
3-13-4 New voicemail message or voice mail status notifications will be delivered to all user device(s) whether at home or roaming

3.13.4 Technical Realisation

The RCS VV-Mail feature requires the Voice Mail System (VMS) to use CMS as message storage to store the voicemail specific user data. The VMS interacts with the RCS client(s) through the Messaging Server for new voicemail message notifications and message status change notifications.

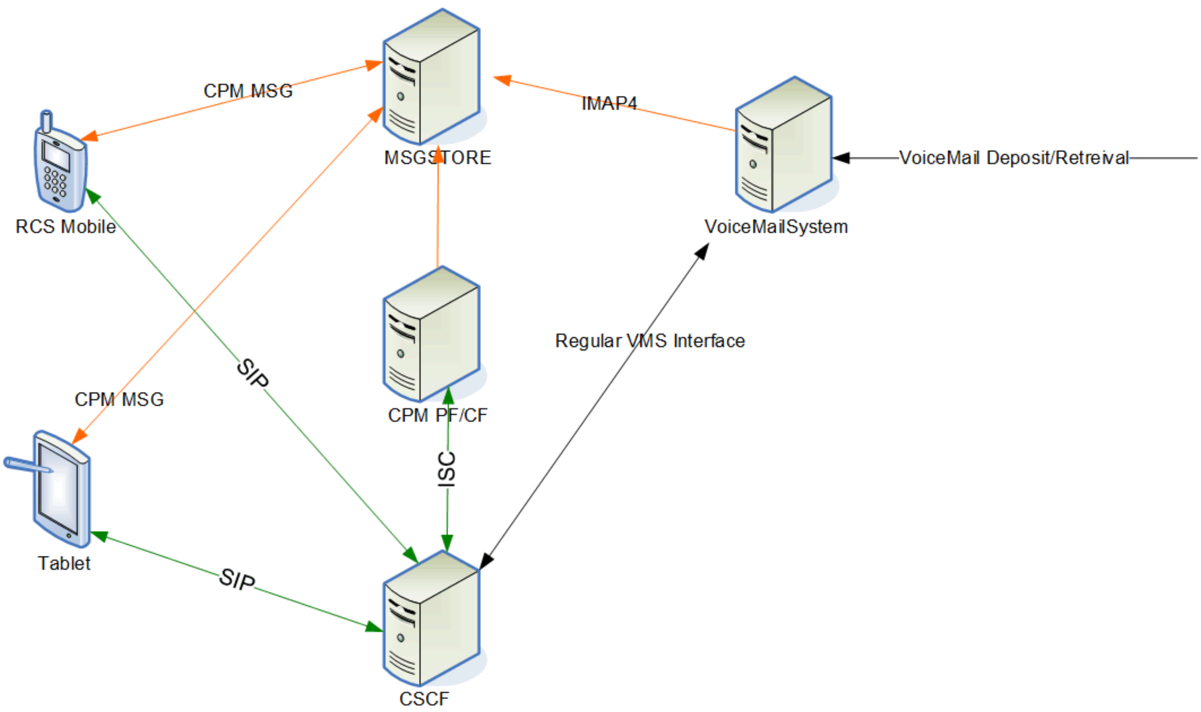

Figure 111: RCS VV-Mail architecture

3.13.4.1 Client configuration

An RCS client can be configured to enable or disable the RCS VV-Mail service based on the operator policy. On the primary device the user could choose to use either the traditional VVM client or the RCS client for the voicemail messaging during the client configuration.

Secondary devices are configured to enable the VMS sync (see section 3.13.4.3.5) when the primary device is using the traditional VVM service for voicemail messaging. When the primary device changes the voicemail messaging mechanism to traditional VVM from RCS VV-Mail, then the corresponding secondary device(s) client configuration parameter RCS VV-MAIL VMS SYNC should be enabled so that the secondary device(s) register with the *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.vmssync"* feature tag.

NOTE: The secondary device configuration changes due to the primary device voicemail messaging mechanism are outside the scope of this specification.

3.13.4.2 RCS VV-Mail messaging

3.13.4.2.1 Voicemail message deposit

When the guest deposits the voicemail for the RCS user, the VMS creates the voicemail message object described in the [RCS-CPM-MSGSTOR-ENDORS]. The VMS may include the transcript of the voicemail into the voicemail object based on the operator policy. Then VMS stores the object in the CMS and sends a new voicemail Standalone Message notification to the Messaging Server. The Messaging Server shall deliver this notification message to all the user's registered RCS clients.

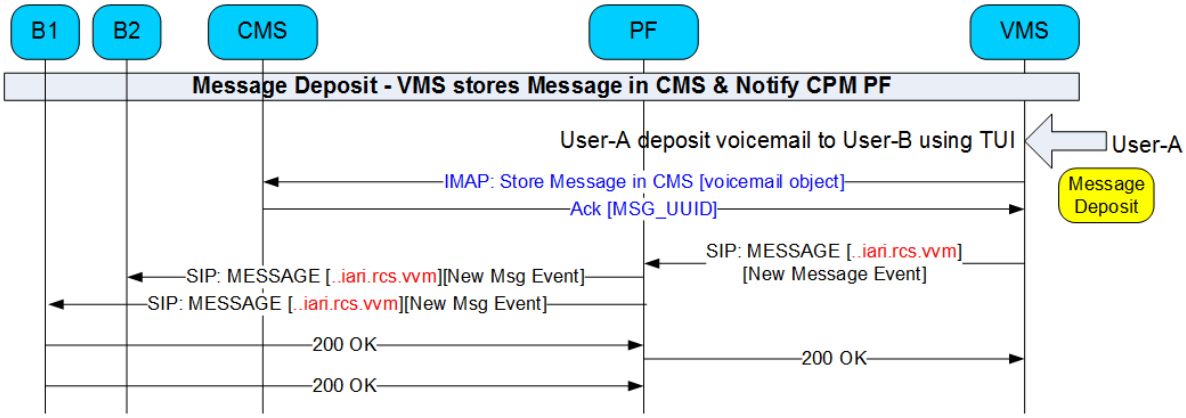

Figure 112: Voicemail message deposit

A Standalone Message is used to notify the new voicemail indication event and includes the feature tag *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.vvm"*, so that the Messaging Server will handle the Standalone Message as required by the RCS VV-Mail feature. When the Messaging Server receives a Standalone Message carrying the new voicemail notification event, it shall fork and deliver the event to all the RCS user's client that are capable of receiving such events (clients registered with the *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.vvm"* feature tag in the Contact header field). If the new message event is not delivered to the primary device, then the Messaging Server shall trigger the interworking as defined in the Standalone Messaging interworking procedures in section 3.2.4.6.1. If the new message event is not delivered to a registered secondary device, then the message is deferred based on operator policy.

The new voicemail message event uses the *application/vnd.gsma.rcs-vvm+xml* content type contained within the CPIM body as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<rcs-vvm xmlns="urn:gsma:params:xml:ns:rcs:rcs:vvm">
        <event-rcs-vvm ev="NM">
              <c>[count of new messages in Inbox]</c>
              <id> [UID of the message]</id>
              <t>[Message type of the new message]</t>
              <s>[identity of guest ]</s>
              <dt>[timestamp of message deposited]</dt>
              <l>[duration of the message]</l>
              <sz>[size of the voicemail message object]</sz>
        </event-rcs-vvm>
</rcs-vvm>
```

Table 58: new voicemail message notification event XML

The tags in the new voicemail message notification event XML data are using the same names as those of the VVM sync sms fields please see section 2.9.1.2 in [VVM] for details on the attributes and values. The message size (voicemail object size) "sz" indicates the size of the message in KB. The message size field is not defined in [VVM].

The field 's' (guest identity) could be populated by VMS as MSISDN, SIP URI or tel URI based on the operator policy.

3.13.4.2.2 Access voicemail message from RCS client

When the RCS client receives the new voicemail message notification, it triggers the VV-Mail folder sync to fetch the new voicemail message(s). The RCS user will be notified about the new voicemail messages including the number of unread messages. The RCS user should be able to access the new voicemail messages from all voicemail entry points (e.g. VVM mailbox, native messaging thread or call log). If the device doesn't have access to the message store, then user can access their voicemail using the TUI.

3.13.4.2.2.1 Client procedures

Once the voicemail messages are accessed from RCS client, the user should be able to manage the messages e.g. listen, call back, delete or save the message. The support of the text transcript of the voicemail messages is optional based on the operator policy. When the transcript is included, the RCS user should be able to read the voicemail message transcript. The RCS client should display the voicemail transcript based on the user local setting.

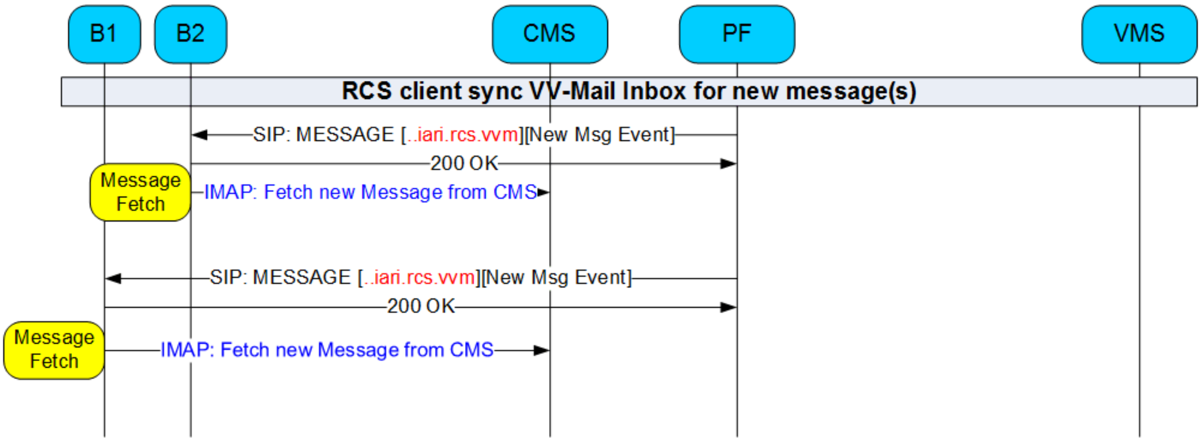

Figure 113: RCS VVM New Message Retrieval

3.13.4.3 Multidevice sync

When the voicemail message is either read or deleted from the RCS client or from the TUI all the user's RCS devices should be synchronised with the message status immediately. For this the RCS clients and the VMS will notify the CPM IMAP event for message read and deleted using the CPM event notification framework as defined in section 4.1.4.8. The RCS client and the VMS will include the feature tag *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.mdsync"* in the *Accept-Contact* header field so that the Messaging Server can fork the event to all other RCS devices of the user.

3.13.4.3.1 Listen to voicemail message from RCS client

RCS user can access voicemail messages from any RCS client and listen to the message. When the message is read, the client should generate a CPM IMAP event to update the message status in the message store. The RCS client should also include the *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.mdsync"* feature tag in the *Accept-Contact* header field so that Messaging Server can fork the event to all other registered RCS clients of the user.

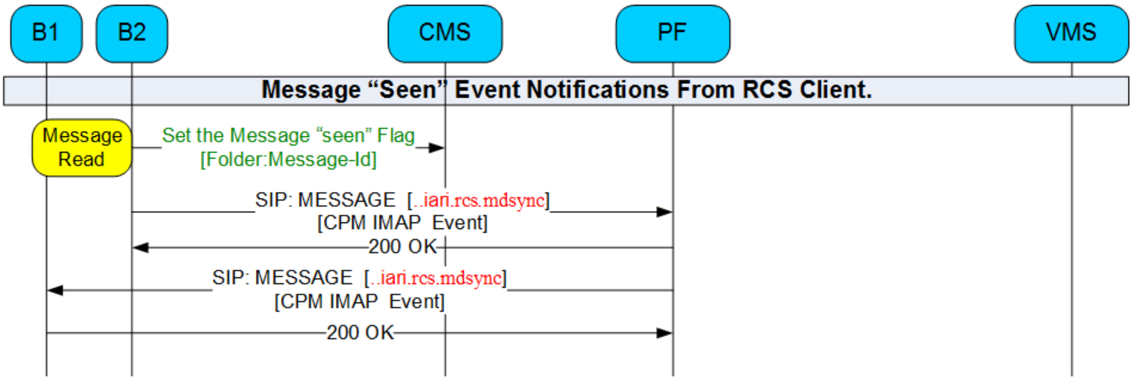

Figure 114: Message listening from the RCS client

3.13.4.3.2 Listen voicemail message from TUI

An RCS user can also access voicemail messages from TUI (e.g. when there is no data connectivity) and listen to the message. When the message is read using the TUI, then the VMS should generate a CPM IMAP event to update the message status in the message store. The VMS should also include the feature tag *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.mdsync"* in the *Accept-Contact* header field so that Messaging Server can fork the event to all other registered RCS clients of the user.

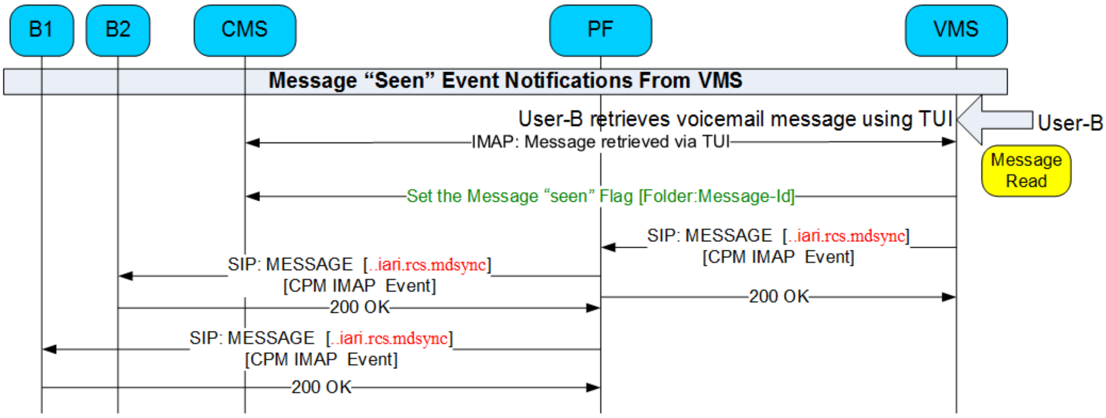

Figure 115: Message listening from the TUI

3.13.4.3.3 Delete voicemail message from RCS client

An RCS user can access voicemail messages from any RCS client and delete the message. When the message is deleted, the client should generate a CPM IMAP event to update the message status in the message store. The RCS client should also include the *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.mdsync"* feature tag in the *Accept-Contact* header field so that Messaging Server can fork the event to all other registered RCS clients of the user. The Messaging Server shall mark the deleted message by setting the 'delete' flag in the 'Inbox' folder.

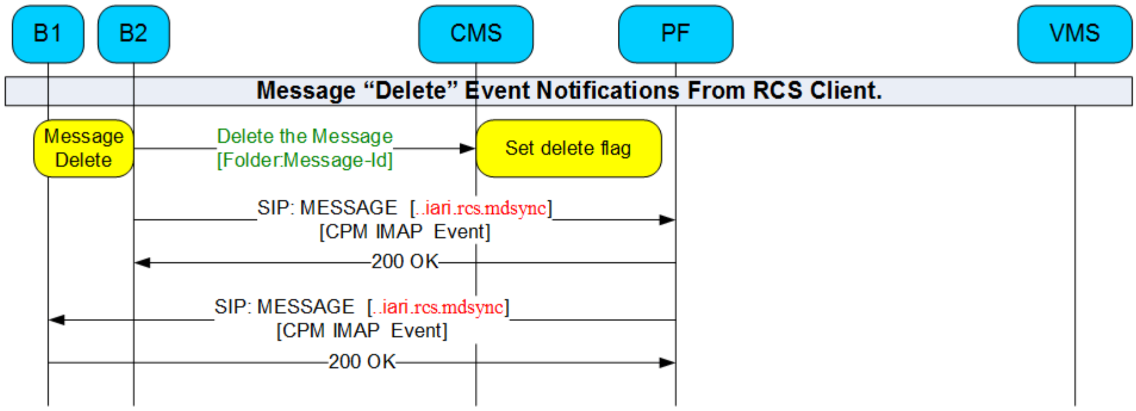

Figure 116: Delete voicemail message from the RCS client

3.13.4.3.4 Delete voicemail message from TUI

An RCS user can also access voicemail messages from the TUI (e.g. when there is no data connectivity) and delete the message. When the message is deleted using the TUI, then the VMS should generate a CPM IMAP event to update the message status in the message store. The VMS should also include the feature tag *+g.3gpp.iari-ref="urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.mdsync"* in the *Accept-Contact* header field so that Messaging Server can fork the event to all other registered RCS clients. The Messaging Server shall mark the deleted message by setting the 'delete' flag in the 'Inbox' folder.

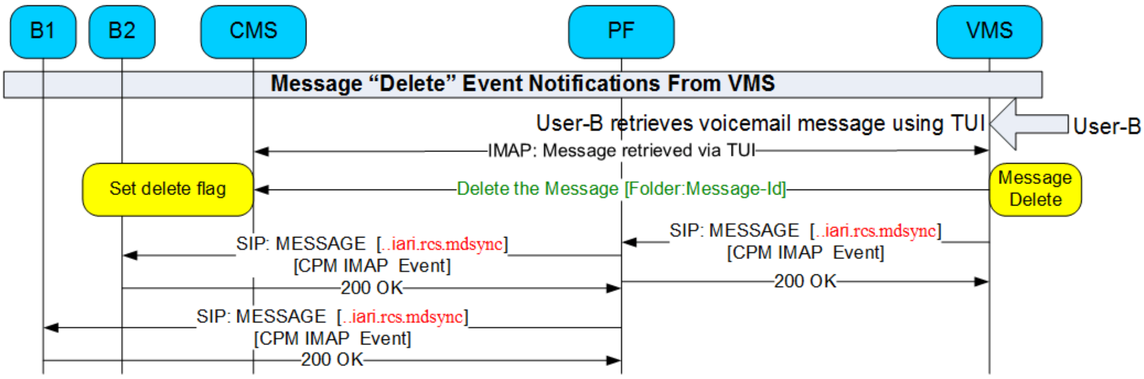

Figure 117: Delete voicemail message from the RCS client

3.13.4.3.5 RCS VV-Mail and VMS sync

The VMS provides the voicemail messaging service also for the traditional VVM. When the primary device is using the traditional VVM service while the secondary device(s) are using the RCS VV-Mail service, then the VMS and RCS VV-Mail needs to be synchronised for a better user experience. When an RCS client on secondary device either reads the voicemail or deletes the voicemail and generates the voicemail CPM IMAP event, it should also generate the VMS sync event based on the RCS VV-MAIL VMS SYNC configuration for the corresponding event when the client is configured for VMS synchronisation. The VMS should then synchronise the message read or delete status with the VVM client (on primary device).

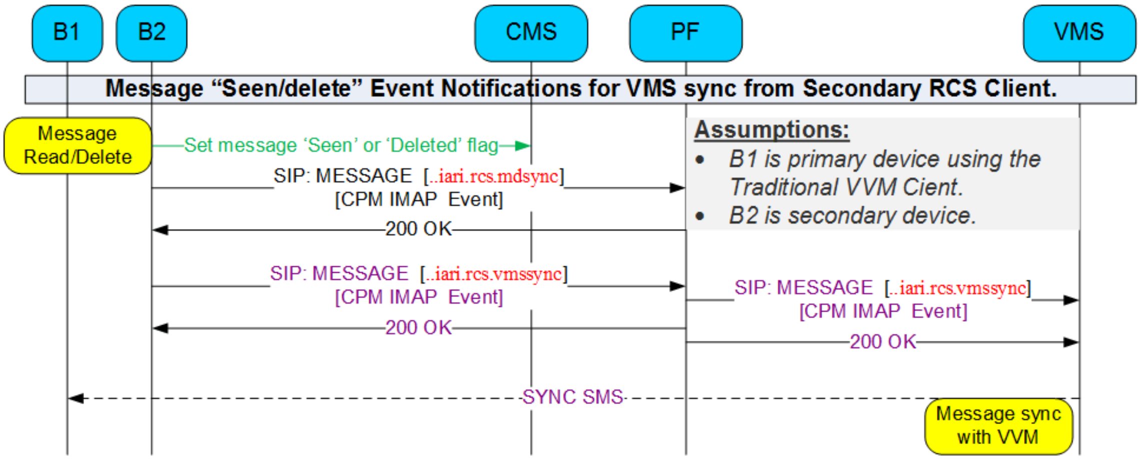

Figure 118: RCS VV-Mail and VMS sync

3.13.4.4 Client Side Spam/Black List Handling

An RCS client should not block new message notification(s) as a result of a local black list.

3.13.4.5 Common Message Store

An IMAP based Common Message Store is expected to store voicemail messages, as described in [RCS-CPM-MSGSTOR-ENDORS]. When the RCS VV-Mail is deployed the voicemail application specific folder shall be created as described in [RCS-CPM-MSGSTOR-ENDORS]. Both the VMS and the RCS client have controlled access to the voicemail folders.

3.13.4.5.1 VMS access to CMS for voicemail functionality

The VMS shall have the write permission access to store the new voicemail message into the CMS and the VMS shall have the read permission access to read the voicemail messages to play them to the users via TUI. The VMS updates the message read and deleted status directly. The VMS should support the message re-save feature via TUI access by changing the 'delete' flag of the messages maintaining the same message properties.

3.13.4.5.2 RCS client access to CMS for voicemail functionality

RCS clients on user devices shall synchronize with the voicemail folders to provide a consistent user experience across all RCS devices. The RCS client will also set the message flags.

3.13.4.5.2.1 Message Re-save from the RCS client

Messages can be marked as deleted are purged based on the operator policy. The RCS client should provide the ability to re-save a deleted message that is not yet purged. When the message is re-saved, the message should maintain the same message properties (e.g. the message date, Id etc...) and the message delete flag shall be reset.

3.13.4.5.2.2 Message archiving

An RCS user can archive the voicemail messages. When voicemail messages are archived, the client will set the Archived flag as described in section 4.1.6.3.

3.13.4.6 Interworking to legacy messaging service

When RCS VV-Mail is deployed, the Standalone Messaging service is used for new message notifications to RCS clients that include the RCS VV-Mail feature tag. The new voicemail notification Standalone Message interworking is applicable as defined in the [RCS-CPM-IW-ENDORS].

NOTE: When the users primary device is registered for service without the RCS VV-Mail service feature tag *+g.3gpp.iari-ref=”urn%3Aurn-7%3A3gppapplication.ims.iari.rcs.vvm"*, the Messaging Server is not required to trigger message interworking.

3.13.5 NNI and IOT considerations

Since RCS VV-Mail service is relevant only to the UNI, no specific NNI guidelines apply.

3.13.6 Implementation guidelines and examples

3.13.6.1 RCS VV-Mail entry points

From a UX point of view, three possible entry points to the RCS VV-Mail service are:

1. Common VM mailbox folder view similar to the traditional VVM view from the RCS client.
2. Native messaging thread like the conversation thread in the context of user contacts.
3. Call log view where the voicemail events shown in the order they received alongside of the call log.

3.13.6.2 Migration of voicemail messages to CMS

When the user upgrades from non-RCS VV-Mail service to RCS VV-Mail service, all the existing voicemail messages stored in the VMS should be migrated to the CMS to provide RCS Messaging Client access to previous messages.

NOTE: Migration and provisioning of system to start using the CMS storage on a per subscriber basis is out of scope of this specification and it is implemented based on the operator discretion.

3.13.6.3 New voicemail notification handling

When the Messaging Server receives the new voicemail notification message, its service behaviour is based on the following:

- The user’s primary device registered and did not register with the RCS VV-Mail service, and then the new message notification is not notified to the primary device.
- The user’s primary device registered with the RCS VV-Mail service and new message notification was not delivered, then the interworking to legacy message delivery is triggered.
- When the Messaging Server doesn’t know about the primary device status and capabilities, then the interworking to legacy message delivery is triggered.
- If there are any secondary devices registered with RCS VV-Mail service, then the new voicemail notification will be delivered.

NOTE: SIM change, SIM swap etc. are out of the scope of this specification and are implemented based on the operator discretion.

4 Cross-service functionality

4.1 Common Message Store

The conversational view of the messaging is used in RCScan be synchronized between a user's multiple devices, by making use of the CPM Common Message Store as described in [RCS-CPM-MSGSTOR-ENDORS].

4.1.1 Feature Description

In RCS, a Common Message Store is used to store messages (standalone text or multimedia and chat messages). An RCS user will have control over the messages to be stored in their Message Store. The Common Message Store allows a user to improve their organization of their stored messages. In addition to this, the Common Message Store is used to provide storage for all messages sent and received by a client supporting the RCS text and multimedia messaging service which also includes any other messages that they receive.

The RCS Common Message Store supports synchronisation of stored objects with the local storage in all registered RCS devices

The storage is always subject to operator-controlled message size and storage quota limitations.

Relevant storage usage information can be collected to allow a service provider to apply usage based charges.

4.1.2 Interaction with other RCS features

The Common Message Store can store messages from all RCS messaging Enablers:

- SMS
- MMS
- RCS Standalone Messaging
- RCS 1-to-1 Chat
- RCS Group Chat
- RCS File Transfer
- RCS Audio Messaging
- RCS Geolocation Push
- RCS Visual Voice Mail

4.1.3 High Level Requirements

This section contains the Common Message Store feature's high level requirements. These requirements are listed in two separate support aspects for client and server as follows:

4.1.3.1 Client/device support

4-1-1 Common Message Store capability: The ability for RCS users to store and manage their messages if the Common Message Store is deployed and the user has a subscription to the Common Message Store.

4-1-2 Event reporting between the RCS client and the Messaging Server to inform of flag update operations on message objects.

4.1.3.2 Server support

4-1-3 Common Message Store capability: For storing a user's messages and synchronising them across RCS user's multiple devices.
4-1-4 Messaging Server supports receiving IMAP flag update events from the RCS client, to be performed on message objects stored in the Common Message Store.

4.1.4 Technical Realisation

RCS supports a "Common Message Store" as described in Section 5.5 of [CPM-SYS_DESC] and specified in [RCS-CPM-MSGSTOR-ENDORS]. Using an IMAP connection, an RCS user can access and manage their stored objects, as described in [RCS-CPM-MSGSTOR-ENDORS] regardless of their RCS service registration.

Regarding the client synchronisation mechanism that applies, client synchronization guidelines are described in section 4.1.6.8.

A primary device sending or receiving messages via the SMS or MMS (e.g. in case of no data connection) may, subject to Service Provider policy regarding automatic SMS or MMS storage in the CMS, also receive these messages via the synchronisation from the Common Message Store. Since legacy messages do not contain Conversation-ID and Contribution-ID a different mechanism is required to link together the two representations of the same message.

Sections 4.1.4.2, 4.1.4.3, 4.1.4.4 and 4.1.4.5 describe the mechanism used for a device to correlate legacy SMS/MMS messages with the same messages already stored in the Common Message Store.

4.1.4.1 Support of GBA in the Common Message Store

The General Bootstrapping Architecture (GBA) defined in [3GPP TS 33.220] provides mechanisms for AKA based user authentication using the 3GPP Authentication Centre (AuC) and the USIM or ISIM. The Common Message Store supports the authentication of primary devices via GBA with the extension defined in this section.

The use of the GBA is based on an existing bootstrapped security association managed between the client and the Bootstrapped Security Function (BSF) operated by the Service Provider. The bootstrapped security association provides the client with a Bootstrapping Transaction Identifier (B-TID) and key material which can be used to authenticate the user with the Service Provider's network.

The client shall use the bootstrapped security association for the authentication with the Common Message procedure if the client configuration parameter MESSAGE STORE AUTH value is set to "2" as defined in Table 75, section A.1.4.3.

In this case the client shall use the key material received from the Service Provider's BSF, the user's private identity (IMPI) and the FQDN of the Common Message Store derived from the value of the configuration parameter MESSAGE STORE URL (see Table 75 section A.1.4.3) to calculate the Ks_NAF as defined in [3GPP TS 33.220]. If no bootstrapped security association exists the client shall first create one as defined in [3GPP TS 33.220].

The client shall login to the Common Message Store either via IMAP AUTHENTICATE command (with PLAIN argument) or IMAP LOGIN command using the B-TID as username and the Ks_NAF as password. The Common Message Store authenticates the request in co-operation with the service Provider's BSF. It returns an OK response to the client if the authentication is successful. If the Common Message Store returns a NO response indicating that the authentication failed then the client shall renegotiate the bootstrapped security association as defined in [3GPP TS 33.220] and [3GPP TS 24.109]. It shall re-attempt the login to the Common Message Store with username and password derived from the B-TID and key material resulting from the re-negotiation.

4.1.4.2 Common Message Store and pager/multimedia-messages

To identify the messages in the Common Message Store that will match legacy SMS/MMS messages sent or received by the device via legacy means, it shall be possible to keep information about the submission or delivery path (SMS or MMS) for converted messages in the Common Message Store.

The information shall be stored for messages by means of the message context for internet mail (see [RFC3458]).

In RCS the following values of the "message-context" are applicable:

- For received messages:
  - pager-message: the message is delivered to a primary device via SMS or stored as SMS
  - multimedia-message: the message is delivered to a primary device via MMS

NOTE: In RCS the message context is only used in the relation between the terminating CPM Participating Function, the Common Message Store and the recipient user's device. It does not provide information of the message context on the originating side nor on the NNI.

- For sent messages:
  - pager-message: the message was sent via SMS or stored as SMS
  - multimedia-message: the message was sent via MMS

4.1.4.2.1 Client initiated storage of SMS/MMS

The RCS Client stores messages in the Default folder.

NOTE: This procedure is applied when synchronising as described in section 4.1.6.8.

When storing a SMS or MMS message and their delivery and read reports in the Common Message Store the client shall follow the definitions for recording of messages in sections 4.1.4.6 and 4.1.4.7 respectively.

If the SMS MESSAGE STORE or MMS MESSAGE STORE configuration parameter defined in Table 75 is set to “always store in the Common Message Store" (i.e. 2) then if storing them locally the client shall upload new sent or received messages and for sent messages, their delivery and read reports (SMS or MMS respectively) to the Common Message Store and link them with the local message. When storing a short message in the Common Message Store the client should not set the correlation information defined in section 4.1.4.3

for SMS. When storing a multimedia message, a delivery or read report in the Common Message Store the client shall set the correlation information defined in section 4.1.4.3 for MMS.

If the SMS MESSAGE STORE or MMS MESSAGE STORE configuration parameter is set to “store if not found in the Common Message Store” (i.e. 1) then the client shall apply the message correlation for new sent or received messages and their delivery and read reports if storing them locally as described in section 4.1.4.3 for SMS and MMS respectively. If the client determines that a locally stored message and its delivery and read report (for sent SMS or MMS respectively) is not already stored in the Common Message Store, the client shall

- store this message to the Common Message Store with the correlation information defined in section 4.1.4.3 (for SMS and MMS) respectively, and
- Link it with the local message.

The client shall store SMS and MMS messages and their delivery and read reports in the Common Message Store in a folder identified by the identity of the Contact in the Conversation under the Default folder. If this folder is not yet created in the Default folder, the client shall first create it. For definition of the folder names refer to [RCS-CPM-MSGSTOR-ENDORS].

When storing a message sent or received as an SMS/MMS in the Common Message Store, the client shall set

- a Message-Context header to pager-message/multimedia-message as described in section 4.1.4.2 and
- The Message-Direction header as described in section 4.1.6.9.

4.1.4.3 Correlating SMS/MMS messages with messages stored in the Common Message Store

The following mechanisms describe how to correlate messages received via legacy means with messages stored in the Common Message Store.

For SMS messages:

- The entity in the network storing the message shall store the prefix of the SMS text body (as defined in section 4.1.4.4) in the Message-Correlator header of the message (see [RCS-CPM-MSGSTOR-ENDORS]).
- The device shall use this Message-Correlator header value, along with To/From headers to find the corresponding locally stored SMS message
- The algorithm is as described in section 4.1.4.4.

For MMS messages:

- Each MMS message and the corresponding delivery reports are defined by a unique MMS Message ID
- The entity in the network storing the message and corresponding delivery reports shall store the MMS Message ID in the Message-Correlator header defined in [RCS-

CPM-MSGSTOR-ENDORS]. The header value shall contain the Message-ID-value defined in [MMSENC] as "ascii-value".

- The device uses the MMS Message ID from MMS messages and delivery reports to find the Unique Identifier (UID) of corresponding MMS messages or delivery notifications in the Common Message Store by matching it with the Message-ID contained in the Message-Correlator of each stored messages.

Since the Common Message Store remains the master storage for these legacy messages, it is up to the client implementation whether or not to discard matched messages received via legacy means.

While correlation collisions will generally be infrequent, there are particular circumstances where they are quite likely to occur. Therefore, in addition to this basic process, additional logic is required to handle correlation collisions for SMSs, see section 4.1.4.5.

4.1.4.4 Correlation Algorithm for SMS

In order to ensure the message correlation algorithm succeeds on both the client and server, RCS clients or the entity in the network sending SMS messages with characters from the GSM 7 bit national single shift and locking shift tables, shall instead use UCS2 (2-byte Universal Character Set) encoding.

NOTE: The RCS client may still receive SMS messages with characters from the GSM 7 bit national single shift and locking shift tables since they may come from non-RCS compliant clients or networks.

When other alphabets (e.g. Latin-1, HP Roman-8) are used as consequence of the SMS bearer technology in the network such as SMPP, the entity creating the Message-Correlator converts the message from the received alphabet into UTF-8 format. The entity which converts from UCS2 or GSM 7 bit alphabets to other alphabets (e.g. Latin-1, HP Roman-8), or vice versa, shall ensure a one-to-one character mapping. This entity can be the entity that stores the message and creates the Message-Correlator, or it can be the entity that sends the message towards the recipient.

NOTE: The one-to-one character mapping is vendor dependent when anything other than GSM 7 bit default alphabet or UCS2 is received by the SMS-C.

The correlation is based on the following field values:

- To: It should be the format as taken from the address field defined in [3GPP TS 23.040]. If TON (Type Of Number) indicates “international”, then a “+” is inserted before the number string. If TON indicates “unknown” only the number string is used. If the address is “alphanumeric”, then the address shall be encoded to UTF-8 format
- From: It should be the format as taken from the address field defined in [3GPP TS 23.040]. If TON indicates “international”, then a “+” is inserted before the number string. If TON indicates “unknown” only the number string is used. If the address is “alphanumeric”, then the address shall be encoded to UTF-8 format.
- The Message-Correlator header value which is generated from the Text Payload contained in the user data of the short message with up to 160 characters as defined below. Characters or data contained in SMS user data information elements (i.e. SMS and EMS control data as well as EMS content data) are not considered for the

correlation algorithm.
Entities storing the message and clients correlating messages shall compose the Message-Correlator header value as follows:

- For messages with no text payload in the SMS user data a Message-Correlator header with no value shall be generated.
- The entity creating the Message-Correlator converts from its original encoding (GSM 7 bit default alphabet or UCS2, see [3GPP TS 23.038], or any other GSM 7 bit national single shift and locking shift tables) into UTF-8 format. The same applies when alphabets other than GSM 7 bit or UCS2 are used (e.g. Latin-1, HP Roman-8).
- Any UTF-8 "Null" character is removed.
- Any UTF-8 characters "CR" and "LF", and the sequence "CR LF", are all removed
- In the case of concatenated SMS messages once the message is reassembled and the above rules have been applied, only the first 160 characters shall be used to generate the Message-Correlator header value in accordance with the procedures defined above.
- If the resulting string contains only US-ASCII characters (0x20 – 0x7e) it will be taken as the value of the Message-Correlator header.
- If the resulting string contains at least one non US-ASCII character, the Message-Correlator header value shall be encoded as defined in [RFC2047]. The value shall be encoded by the use of the UTF-8 character set (charset = utf-8) and base64 encoding (encoding = b). In this case the client should use for correlation of messages the "encoded-text" part of the header value. For details of the Message-Correlator header encoding refer to [RCS-CPM-MSGSTOR-ENDORS].
- Examples of Message-Correlator header values:
  the Message-Correlator header value of a short message with the text payload:
  `To your health, my friend`
  will encoded as follows
  `Message-Correlator: To your health, my friend`
  the Message-Correlator header of a short message with the text payload
  `На здоровье, мой друг`
  will be encoded as follows
  `Message-Correlator: =?utf-8?b?`
  `0J3QsCDQt9C00L7RgNC+0LLRjNC1LCDQvNC+0Lkg0LTRgNGD0LM=?=`

Table 59 illustrates the required coding conversions for the Message-Correlator algorithm to succeed.

ID	Messaging Technology	Original Encoding type	Client Sender		Client Receiver	Messaging Server is Receiver from SMS-C via SMPP or MAP
			Encoding for Sending Message	Calculation of Message-Correlator	Calculation of Message-Correlator	Calculation of Message-Correlator
1	Legacy SMS	GSM 7 bit default alphabet	GSM 7 bit default alphabet	Convert original message text payload into UTF-8	Convert received message text payload into UTF-8	Convert received message text payload into UTF-8
2		UCS2	UCS2	Convert original message text payload into UTF-8	Convert received message text payload into UTF-8	Convert received message text payload into UTF-8
3		Other alphabets using an 8 bit encoding (e.g. Latin-1, HP Roman-8)	Not applicable [1]	Not applicable[1]	Not applicable[1]	Convert received message text payload into UTF-8. The entity which converts from UCS2 or GSM 7 bit alphabets to other alphabets (e.g. Latin-1, HP Roman-8) shall ensure a one-to-one character mapping
4		Message with GSM 7 bit national alphabet (single shift and locking shift tables)	UCS2	Convert original message text payload into UTF-8	Convert received message text payload into UTF-8	Convert received message text payload into UTF-8

ID	Messaging Technology	Original Encoding type	Client Sender		Client Receiver	Messaging Server is Receiver from SMS-C via SMPP or MAP
			Encoding for Sending Message	Calculation of Message-Correlator	Calculation of Message-Correlator	Calculation of Message-Correlator
5		Message with GSM 7 bit national alphabet (single shift and locking shift tables)	GSM 7 bit (including shift tables[2]) (used by legacy SMS clients that are not RCS compliant)	Original message to UTF-8 using the GSM 7 bit national single shift and locking shift tables[2]	Convert received message text payload into UTF-8 using the GSM 7 bit national single shift and locking shift tables[2]	Convert received message text payload into UTF-8. Support of GSM 7 bit national single shift and locking shift table is required by the entity generating the Message-Correlator. NOTE: if a network entity is involved which converts this to another alphabet (e.g. Latin-1, HP Roman-8), a one-to-one mapping may not be possible and thus the Message-Correlator algorithm would not succeed.
6	Standalone Message (when stored as SMS)	UTF-8	UTF-8	When stored as SMS, use original message text payload which is already in UTF-8	When stored as SMS, use original message text payload which is already in UTF-8	Convert received message text payload into UTF-8. NOTE: For standalone messages with Pager Mode, it is possible based on Service Provider policies, for these messages to be automatically stored in the CMS as legacy SMS messages. In this case the client will not be able to correlate these messages with their copy in the Common Message Store based on the procedures for matching of messages defined in [RCS-CPM-CONVFUNC-ENDORS]. Since the client is not able to correlate the standalone messages with Pager Mode based on the procedures defined in [RCS-CPM-CONVFUNC-ENDORS] it shall then attempt to correlate them as per this row, i.e using Message-Correlator. See section 4.1.6.8.

Table 59: Encoding conversions for Message-Correlator algorithm

NOTES to Table 59:

1. A client encoding SMS messages using 8 bit encoding alphabets (e.g. Latin-1, HP Roman-8) instead of GSM 7 bit or UCS2 or a client sending encrypted messages over SMS are not supported by the current algorithm.
2. Correlation of SMS messages using national single shift and locking shift tables indicated in the user data header Info Element is not fully supported, especially if either the client or the entity in the network has no access to the SMS PDU or does not support the used national single shift or locking shift tables. Therefore an RCS Client shall encode SMS messages as UCS2.

Additional considerations:

- For the correlation of outgoing messages the From field is not used
- For the correlation of incoming messages the To field is not use
- The correlation is achieved by Message-Correlator header value, using a case-sensitive comparison.
  The matching algorithm should take into account differences in the presentation of the address string according to different types of numbers.
  The creation of a Message-Correlator header value used for the correlation via a full string match requires in some scenarios access to the native SMS Transfer Protocol Data Units (TPDU, i.e. the TP-UD, TP-DCS data units). Client implementations that do not have access to the TPDU but only to the "interpreted" payload of the short message or if the message contained characters encoded via single or locking shift tables may compensate for this by using alternative matching algorithms which are out of scope for this specification.

4.1.4.5 Dealing with Collisions

The correlation field values are used to correlate between SMS messages on the Common Message Store and on the device. Specifically, when the device synchronises with the Message Store Server it will obtain UIDs and the correlation field values for those SMS messages that are new or have changed since the last synchronisation. The device will then attempt to correlate the UIDs and correlation field values with any messages it has received or subsequently receives from the network. Therefore, if any of the messages have the same correlation field values (this is considered a correlation “collision”) then the device cannot distinguish between them when matching to its local messages.

The device should compare the direction (originating or terminating) in addition to comparing the correlation field values, meaning that correlation collisions can only occur on messages with the same direction.

Correlation collisions can occur in these two cases:

1. Messages in the same thread with the same content, typically when they are chronologically close (so returned on the same synchronization) SMS messages in the same thread with the same content, such as successive replies both saying “OK”.
2. Messages in the same thread with content that is different only after the first 140 bytes. This is more likely when higher numbers of messages are being compared, for

example, a likely worst case example would be when a phone has been switched off for a long period (e.g. a vacation, a repair). This rare scenario is not addressed here further.

If there are collisions, the device should identify the chronologically first received message on the device with the lower UID on Message Store Server.

For example, suppose Message Store Server returns two new messages both with the same value C for the Message-Correlator header but with UIDs x and y, x < y, and the device has received two messages with the same value C for the Message-Correlator header at times t1 and t2, t1 < t2. Then the device should identify t1 = x and t2 = y.

The same principle applies when the number of correlation collisions on the device is different from the number on the Message Store Server; those are usually cases of temporary lack of synchronisation between the device and the Common Message Store.

As an example, suppose as above the Common Message Store has the same two new messages but the device has only received one message with value C for the Message-Correlator header. It should identify that with UID x, in the presumption that the network will shortly deliver a second message with value C for the Message-Correlator header which it will then identify with UID y. Similarly, if the Common Message Store only has UID x producing value C for the Message-Correlator header but the device has both t1 and t2, the device should identify t1 with message x and expect a subsequent synchronisation to return message y which it will then identify with t2.

Note that some legacy messages might not have been stored in Common Message Store by the network. Therefore the length of time between the messages should be considered by the client when determining whether the messages are duplicates. Note also that the device would have to take into account messages the device might have that it received before the Common Message Store was in place.

The impact of correlation collisions in this method may result in a wrong correlation; in the case above, to identify t1 = x and t2 = y when the correct mapping was in fact t1 = y and t2 = x. In this case, the view from one device and another will be out of sync: a user making a state change to t1 on one device will see it applied to t2 on the other device, when they would expect it to apply to message y.

For example, take the case of successive identical messages. If the user marks on one device the earlier of these messages as a favourite, then the device view might be as follows:

“are you still on for tonight?”
“yes” <- FAVOURITE
“do you have the tickets?”
“yes”

whereas on another device the view would be:

“are you still on for tonight?”
“yes”
“do you have the tickets?”

"yes" <- FAVOURITE

No messages are lost, so there is no need to define any more advanced methods.

4.1.4.6 Recording of SMS messages

Apart from text transfer SMS provides a number of enhanced messaging capabilities as well as device and service control functions. For the synchronisation of SMS messages across devices it is essential that the client and the Common Message Store assume a common rule for SMS message recording.

Short messages are recorded in the Common Message Store either by the network or by the device based on the definitions in section 4.1.4.2.1. Entities recording short messages need to follow these guidelines.

NOTE: If the client has no access to the SMS PDUs, it may assume that all locally received messages that it has access to fulfil the rules for recording.

Clients matching received messages with a message in the Common Message Store need to consider these guidelines to apply matching only for messages that are subject to recording.

For clients fetching messages from the Common Message Store to update the local storage no special considerations are required.

The following PDU types need to be recorded for mobile originated SMS:

- SMS-SUBMIT
- SMS-STATUS-REPORT

The following PDU types need to be recorded for mobile terminated SMS:

- SMS-DELIVER

For a definition of the SMS PDU types refer to [3GPP TS 23.040].

4.1.4.6.1 Recording of SMS-SUBMIT and SMS-DELIVER

A short message may consist of multiple parts, the content transferred in the actual user data and the content of the user data headers inserted in the user data. The following sections define the recording rules for the two parts of the message.

4.1.4.6.1.1 User Data

The User Data is the field of a Short Message PDU that carries the user content (see [3GPP TS 23.040]). This section provides the rules for recording of messages based on the values of the SMS Data Coding Scheme and Protocol Identifier fields [3GPP TS 23.040].

4.1.4.6.1.2 SMS Data Coding Scheme

The SMS Data Coding Scheme field of the short message indicates the encoding used for the user data.

The indication of text compression and character set are used by the recording entity to decode the message content. The message is not stored in the Common Message Store if the Character Set indicates "8 bit data". In all other cases the message is converted in text using the UTF-8 character set for storage. If the User Data is the only content of an SMS message it is stored in the Common Message Store the body of a CPIM message.

The SMS Class determines the routing of a mobile terminated short message on the device. Messages without a Message Class shall be recorded based on the principles defined in section 4.1.4.2.

A mobile terminated message indicating Class "0" shall not be recorded in the network. It is sent to the mobile device via SMS. The device will display the message immediately without storing. If the user decides to store the message locally on the device after display the client shall upload the message to the Common Message Store without applying the Correlation Mechanism defined in section 4.1.4.2. The client shall store the message under the Default folder identified by the identity of the Contact in the Conversation. If this folder is not yet created in the Default folder, the client shall first create it. The client shall store the message with a Message-Context header set to pager-message as described in section 4.1.4.2.

Mobile terminated messages with Message Class "1", "2" and "3" shall be recorded by the network or the client unless content encoding (e.g. "8 bit data") or the high layer protocol indication prohibit recording.

If the Data Coding Scheme indicates a message for automatic deletion after reading it shall not be recorded.

Messages with a Message Waiting Indication shall not be recorded.

4.1.4.6.1.3 Protocol Identifier

The Protocol Identifier indicates whether a higher layer protocol is used to encode the content of the User Data. Only messages with a Protocol Identifier set to the default 0x00 and to values from the range 0x81-0x87 (Replace Short Message Type) shall be recorded.

If a Replace Short Message Type is set in the Protocol Identifier of a mobile terminated short message (value in the range of 0x81-0x87), then the client shall apply the handling as defined in [3GPP TS 23.040], i.e. it replaces the content of a stored message that matches the Replace Type of the new message. If the client configuration parameter SMS MESSAGE STORE is set to value "1" or "2" the client shall update the Common Message Store according to the result of the local message processing. If the client stores a message in the Common Message the Replace-Short-Message-Type header value shall be set to the value received in the short message.

The entity in the network storing the message in the Common Message Store shall apply the following handling. If a message with a Replace Short Message Type is received, it shall select the folder where the conversation is stored, as identified by the originator address. It shall search in the folder the messages with the same Replace Short Message Type as the new message.

If a message with the same Short Message Replace Type header is found, the \Deleted flag shall be set for this message. The new message is stored in the selected folder with a Short Message Replace Type header set.

If no related message or folder exists the message is stored in the regular fashion. The Message is always stored with the Replace-Short-Message-Type header value set to the value received in the short message.

4.1.4.6.1.4 Short Message Object

A short message shall be recorded in the Common Message Store with the headers set as defined below.

Mail Header	Status	Content
From	Mandatory	For mobile originated Short Messages it contains the public user identity of the sender. It is either encoded in the global-number representation of a tel URI as defined in [RFC3966] or as a SIP URI as defined in [RFC3261]. For mobile terminated Short Messages the value is derived from the SMS originator address. If the originator is identified by a E.164 number it is encoded in the global-number representation of a tel URI as defined in [RFC3966] If the originator is identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. If the originator is identified by an alphanumeric string, the From field contains the string only.
To	Mandatory	For mobile originated short messages the value is derived from the SMS destination address. If the destination is identified by a E.164 number it is encoded in the global-number representation of a tel URI as defined in [RFC3966] If the destination is identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. For mobile terminated Short Messages it contains the public user identity of the sender. It is encoded in the global-number representation of a tel URI as defined in [RFC3966] or as a SIP URI as defined in [RFC3261].
Date	Mandatory	Indicates the time the message was recorded
Conversation-ID	Mandatory	It shall be assigned by the entity that stores the message
Contribution-ID	Mandatory	It shall be assigned by the entity that stores the message.
IMDN-Message-ID	Mandatory	It shall be assigned by the entity that stores the message.
Message-Correlator	Optional	If present it shall contain the SMS message Correlator for message correlation as defined in section 4.1.4.3. See also section 4.1.4.2.
Message-Context	Mandatory	Message-Context shall be set to "pager-message"
Message-Direction	Optional	Message Direction header value shall be set as defined in section 4.1.6.9.

Mail Header	Status	Content
Message-ID	Optional	It is assigned by the entity that stores the message. If stored, the header value shall conform to the definitions of [RFC5322].
Replace-Short-Message-Type	Optional	Indicates the replacement type of the short message. It can have the values 1 – 7. The value shall be taken from the value of SMS Protocol Identifier (see [3GPP TS 23.040])
Content-Type	Mandatory	Message/CPIM

Table 60: Mail Headers of the Short Message Object

The Message Body of the Short Message Object shall contain the following CPIM headers:

CPIM Header	Status	Content
From	Mandatory	For mobile originated Short Messages it contains the public user identity of the sender. It is either encoded in the global-number representation of a tel URI as defined in [RFC3966] or as a SIP URI as defined in [RFC3261]. For mobile terminated Short Messages the value is derived from the SMS originator address. If the originator is identified by a E.164 number it is encoded in the global-number representation of a tel URI as defined in [RFC3966] If the originator is identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. If the originator is identified by an alphanumeric string, the From field contains the string only.
To	Mandatory	For mobile originated short messages the value is derived from the SMS destination address. If the destination is identified by a E.164 number it is encoded in the global-number representation of a tel URI as defined in [RFC3966] If the destination is identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. For mobile terminated short messages it contains the public user identity of the sender. It is encoded in the global-number representation of a tel URI as defined in [RFC3966] or a SIP URI as defined in [RFC3261].

CPIM Header	Status	Content
DateTime	Mandatory	For Mobile Terminated Messages it should indicate the time the message was received in the Originator Service Centre. Derived from the SMS Service Centre Time Stamp (see [3GPP TS 23.040]). If the SMS Service Centre Timestamp is not available it should contain the time of message recording. For Mobile Originated Messages it should indicate the time the message was sent by the client that recorded the message or the time the message was received in the Originator Service Centre.
imdn.Message-ID	Mandatory	Shall be set to the same value as the corresponding MIME header (see Table 60).
rcs.Message-Correlator	Optional	If present it shall contain the SMS message Correlator for message correlation as defined in section 4.1.4.3. See also section 4.1.4.2.
rcs.Message-Context	Optional	The value of rcs.Message-Context shall be set to "pager-message". It shall be present if the rcs.Message-Correlator header is present. It indicates that the rcs.Message-Correlator contains the SMS message correlation as defined in section 4.1.4.3.
rcs.Service-Centre-Address	Optional	Indicates the address of the Short Message Service Centre used for the transfer of the short message. For mobile originated messages it is derived from the transport destination address. For mobile terminated messages it is derived from the transport originating address. See [3GPP TS 23.040]
rcs.Reply-Path	Optional	Indicates whether a reply-path exists for the message. If the value is set to "1" the reply-path exists, if set to "0" or the header is not present the reply-path does not exist. For a description of the reply-path refer to [3GPP TS 23.040].
rcs.Replace-Short-Message-Type	Optional	Indicates the replacement type of the short message. It can have the values 1 – 7. The value shall be taken from the value of SMS Protocol Identifier (see [3GPP TS 23.040])
Content-Type	Mandatory	For messages with text part only the Message body should be encoded using content-type text/plain in UTF-8 encoding.
Content-Transfer-Encoding	Optional	Typical content transfer encoding shall be used, e.g. quoted-printable or base64

Table 61: CPIM Header of the Short Message Object

Example of a recorded short message:

```
From: ACME Corporation
To: tel:+4917112345678
Date: Tue, 27 01 2015 17:57:16 +0100
Conversation-Id: 4465a75f-25f5-46dc-8da5-773a1ba174fc
Contribution-Id: f592acbb-fef9-489f-84be-e4596b607c6d
IMDN-Message-ID: 654131a654131a654131a654131a8994656
Message-ID: 6159659256@pf.operator.com
```

```
Message-Direction: received
Replace-Short-Message-Type: 1
Message-Correlator: This is the message content.
Message-Context: "pager-message"
Content-Type: Message/CPIM

From: ACME Corporation
To: +4917112345678
NS: imdn <urn:ietf:params:imdn>
NS: rcs <http://www.gsma.com>
imdn.Message-ID: 654131a654131a654131a654131a8994656
rcs.Message-Correlator: This is the message content.
rcs.Message-Context: "pager-message"
rcs.Service-Centre: +49171202020200
rcs.Reply-Path: 0
rcs.Replace-Short-Message-Type: 1
DateTime: 2015-01-27T16:32:55+00:00

Content-Type: text/plain; charset=utf-8

This is the message content.
```

4.1.4.6.1.5 User Data Header

User Data Headers can be used for encoding for SMS control functions as well as special SMS content (see [3GPP TS 23.040]. The type of content of a User Data Header is identified by the Information Element Identifier (IEI) within a User Data header. The following Information Identifiers need to be considered for the processing of short messages.

Concatenated short messages reference numbers IEI need to be considered when processing a concatenated message. Only the re-assembled message shall be recorded in the Common Message Store.

NOTE: The recording requirements for other values of Information Element Identifiers are for further study.

4.1.4.6.2 Recording of SMS-STATUS-REPORT

The SMS-STATUS-REPORT informs the message sender about the status of a previously sent message if it has been requested by the sender. The SMS-STATUS-REPORT shall be used by the entity storing SMS messages in the Common Message Store to record a delivery notification as defined in [RCS-CPM-CONVFUNC-ENDORS].

The SMS-STATUS-REPORT is matched on the originating side to the original sent message by use of SMS Message Reference assigned by the originating device. The SMS Message Reference is not globally unique, thus the matching of SMS-STATUS-REPORT to the original short message across multiple devices has some limitations.

A delivery notification shall be recorded if the Status field in the SMS-STATUS-REPORT indicates that the short message has been "received by the SME" (see [3GPP TS 23.040]). In all other cases a delivery notification shall not be recorded.

A SMS delivery notification shall be recorded in the Common Message Store with mail and CPIM headers set as follows.

Mail Header	Status	Content
From	Mandatory	It contains the address of the recipient of the original message as derived from the recipient address field of the SMS-STATUS-REPORT see [3GPP TS 23.040]. If identified by a E.164 number it is encoded in the global-number representation of a tel URI as defined in [RFC3966] If identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context.
To	Mandatory	It contains the public user identity of the sender. It is encoded in the global-number representation of a tel URI as defined in [RFC3966] or as a SIP URI as defined in [RFC3261].
Date	Mandatory	Indicates the time the message was recorded
Conversation-ID	Mandatory	It shall have the same value as the original sent message.
Contribution-ID	Mandatory	It shall have the same value as the original sent message.
Message-ID	Optional	It is assigned by the entity that stores the message. If stored, the header value shall conform to the definitions of [RFC5322].
IMDN-Message-ID	Mandatory	It shall be assigned by the entity that stores the message.
Content-Type	Mandatory	Message/CPIM

Table 62: Mail Headers of the SMS Delivery Notification

The Message Body of the SMS Delivery Notification shall contain the following CPIM headers:

CPIM Header	Status	Content
From	Mandatory	It contains the address of the recipient of the original message as derived from the recipient address field of the SMS-STATUS-REPORT see [3GPP TS 23.040]. If identified by a E.164 number it is encoded in the global-number representation of a tel URI as defined in [RFC3966] If identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context.
To	Mandatory	It contains the public user identity of the sender. It is encoded in the global-number representation of a tel URI as defined in [RFC3966]
DateTime	Mandatory	It should indicate the time the message was delivered by the Service Centre. The value is derived from the Discharge Time of the SMS-STATUS-REPORT (see [3GPP TS 23.040]). If it is not available then it should contain the time of message recording.
imdn.Message-ID	Mandatory	Shall be set to the same value as the corresponding MIME header (see Table 62).

CPIM Header	Status	Content
Content-Type	Optional	message/imdn+xml

Table 63: CPIM Header of the SMS Delivery Notification

The message body of the CPIM message shall contain an IMDN with status set to "delivered" and the imdn.Message-ID assigned to the original sent message.

4.1.4.7 Recording of MMS messages

MMS provides a rich messaging service for multimedia content to be sent to a single recipient or to a list of recipients.

Multimedia messages are recorded in the Common Message Store either by the network or by the device based on the definitions in section 4.1.4.2.1. Entities recording multimedia messages need to follow these guidelines.

The following PDU types need to be recorded for mobile originated MMS:

- MM1 Submission
- MM1 Delivery Report
- MM1 Read-Reply Report

The following PDU types need to be recorded for mobile terminated MMS:

- MM1 Delivery
- MM1 Read-Reply Report

4.1.4.7.1 Recording of MM1 Submission and MM1 Delivery

The entity storing a MMS message in the Common Message Store shall use the Multimedia Message object defined in this section.

A multimedia message shall be recorded in the Common Message Store with the headers set as defined below.

Note: The encoding of addresses in mail headers of a recorded MMS message follows the SIP encoding principles as defined in [RCS-CPM-MSGSTOR-ENDORS] whereas the address encoding of an MMS is based on the MMS Addressing Model of [MMSENC]. The entity storing a MMS message needs to re-format address headers.

For an MMS Message to multiple recipients the entity storing the message shall store the participant list in a recipient-list-history content part with *copyControl* set in accordance with the recipient address type, To:, Cc: or Bcc:.

Mail Header	Status	Content
From	Mandatory	For mobile originated MMS messages it contains the public user identify of the sender. It is either encoded in the global-number representation of a tel URI as defined in [RFC3966] or as a SIP URI as defined in [RFC3261]. For mobile terminated MMS Messages the value is derived from the MMS originator address. If the originator is identified by a E.164 number it is encoded in the global-number representation of an tel URI as defined in [RFC3966] If the originator is identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. If the originator is identified by an e-mail address it is encoded as mailbox as defined in [RFC5322].
To	Optional	For mobile originated MMS messages the value is derived from the MMS recipient address "To" field. If the recipient is identified by an E.164 number it is encoded in the global-number representation of an tel URI as defined in [RFC3966] If the recipient is identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. If the originator is identified by an e-mail address it is encoded as mailbox as defined in [RFC5322]. For a MMS message to multiple recipients the To header value shall be encoded as [RFC5322] address-list with each address value following the encoding rules above. In addition the multiple recipient addresses shall be represented in a body part of the message containing a recipient-list-history as defined in [RFC5365]. For mobile terminated MMS Messages it contains the public user identity of the recipient. It is encoded in the global-number representation of a tel URI as defined in [RFC3966] or as SIP URI as defined in [RFC3261]. At least one of To, Cc or Bcc header shall be present in the message.

Mail Header	Status	Content
Cc	Optional	For mobile originated MMS messages the value is derived from the MMS recipient address "Cc" field. If the recipient is identified by an E.164 number it is encoded in the global-number representation of an tel URI as defined in [RFC3966] If the recipient is identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. If the originator is identified by an e-mail address it is encoded as mailbox as defined in [RFC5322]. For a MMS message to multiple recipients the Cc header value shall be encoded as [RFC5322] address-list with each address value following the encoding rules above. Multiple recipient addresses shall be represented in a body part of the message containing a recipient-list-history as defined in [RFC5365]. For mobile terminated MMS Messages it contains the public user identity of the sender. It is encoded in the global-number representation of a tel URI as defined in [RFC3966] or a SIP URI as defined in [RFC3261]. At least one of To, Cc or Bcc header shall be present in the message.
Bcc	Optional	For mobile originated MMS messages the value is derived from the MMS recipient address "Bcc" field. If the recipient is identified by an E.164 number it is encoded in the global-number representation of an tel URI as defined in [RFC3966] If the recipient is identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. If the originator is identified by an e-mail address it is encoded as mailbox as defined in [RFC5322]. For a MMS message to multiple recipients the Bcc header value shall be encoded as [RFC5322] address-list with each address value following the encoding rules above. Multiple recipient addresses shall be represented in a body part of the message containing a recipient-list-history as defined in [RFC5365]. For mobile terminated MMS Messages it contains the public user identity of the sender. It is encoded in the global-number representation of a tel URI as defined in [RFC3966] or a SIP URI as defined in [RFC3261]. At least one of To, Cc or Bcc header shall be present in the message.

Mail Header	Status	Content
Date	Mandatory	Indicates the time the message was recorded
Conversation-ID	Mandatory	It shall be assigned by the entity that stores the message
Contribution-ID	Mandatory	It shall be assigned by the entity that stores the message.
Message-ID	Optional	It is assigned by the entity that stores the message. . If stored, the header value should conform to the definitions of [RFC5322]. In a transition period Service Providers may store in the Message-ID header the value of the MMS Message-ID for compatibility with earlier versions of this specification.
IMDN-Message-ID	Mandatory	It shall be assigned by the entity that stores the message. The IMDN-Message-ID header value should generated with similar definition and uniqueness requirements as described in section 6.3 of [RFC5438]. In a transition period Service Providers may store in the IMDN-Message-ID header the value of the MMS Message-ID for compatibility with [RCS-CPM-MSGSTOR-ENDORS].
Message-Correlator	Mandatory	It shall contain the MMS Message-ID for message correlation as defined in section 4.1.4.3. See also section 4.1.4.2
Message-Context	Mandatory	Message-Context shall be set to "multimedia-message"
Message-Direction	Optional	Message Direction header value shall be set as defined in section 4.1.6.9.
Content-Type	Mandatory	Message/CPIM

Table 64: Mail Headers of the Multimedia Message Object

The Message Body of the Multimedia Message Object shall contain the following CPIM headers:

CPIM Header	Status	Content
From	Mandatory	For mobile originated MMS messages it contains the public user identity of the sender. It is either encoded in the global-number representation of a tel URI as defined in [RFC3966] or as a SIP URI as defined in [RFC3261]. For mobile terminated MMS Messages the value is derived from the MMS originator address. If the originator is identified by a E.164 number it is encoded in the global-number representation of an tel URI as defined in [RFC3966] If the originator is identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. If the originator is identified by an e-mail address it is encoded as mailbox as defined in [RFC5322].

CPIM Header	Status	Content
To	Optional	For mobile originated MMS messages the value is derived from the MMS recipient address "To" field. If the recipient is identified by an E.164 number it is encoded in the global-number representation of an tel URI as defined in [RFC3966] If the recipient is identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. If the originator is identified by an e-mail address it is encoded as mailbox as defined in [RFC5322]. For a MMS message to multiple recipients there shall be To and Cc header fields per recipient address. Bcc recipients addresses shall be contained in a To header field. If a client needs to represent the Bcc destination address classification it shall use the header from the mail headers instead. Multiple recipient addresses shall be represented in a body part of the message containing a recipient-list-history as defined in [RFC5365] with a mapping of the address header field type to copyControl parameter. For mobile terminated Short Messages it contains the public user identity of the sender. It is encoded in the global-number representation of a tel URI as defined in [RFC3966] or a SIP URI as defined in [RFC3261]. At least one of To or Cc header shall be present in the message.
Cc	Optional	For mobile originated MMS messages the value is derived from the MMS recipient address "Cc" field. If the recipient is identified by an E.164 number it is encoded in the global-number representation of an tel URI as defined in [RFC3966]. If the recipient is identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. If the originator is identified by an e-mail address it is encoded as mailbox as defined in [RFC5322]. For a MMS message to multiple Cc recipients there shall be a Cc header field per recipient address. At least one of To or Cc header shall be present in the message.

CPIM Header	Status	Content
DateTime	Mandatory	For Mobile Terminated Messages it should indicate the time the message was received in the Originator Service Centre. Derived from the MMS Date header (see [MMSENC]). If the Date header is not available it should contain the time of message recording. For Mobile Originated Messages it should indicate the time the message was sent by the client that recorded the message or the time the message was received in the Originator Service Centre.
Subject	Optional	Indicates the Subject of the MMS message. It shall be taken from the Subject header of the MMS message.
imdn.Message-ID	Mandatory	Shall be set to the same value as the corresponding MIME header (see Table 64).
imdn.Disposition-Notification	Optional	This header indicates whether the sender has requested a MMS read-reply report. The entity recording the message shall derive the value from the MMS X-Mms-Read-Reply header defined in [MMSENC]. If the X-Mms-Read-Reply header is present and set to "yes", the imdn.Disposition-Notification header shall take the value "display". Otherwise the header shall be absent.
rcs.Message-Correlator	Mandatory	It shall contain the MMS Message-ID.
rcs.Message-Context	Mandatory	The value of rcs.Message-Context shall be set to "multimedia-message". It indicates that the rcs.Message-Correlator contains the MMS message correlation (MMS Message-ID) as defined in section 4.1.4.3.
rcs.Mms-Message-Class	Optional	The MMS Message Class indicates the class of the multimedia message. It may take the values "Personal", “Advertisement", "Informational" and "Auto". Multimedia messages with value "Auto" typically contain MMS read report. If rcs.Mms-Class is absent the value "Personal" shall be assumed.
Content-Type	Mandatory	The Content-Type defines the type of content of the body part of the CPIM message which is likely to be a multipart. For the user content the Content-Type shall be taken from the original multimedia message. All types of content allowed for MMS are relevant for storage in the Common Message Store, e.g. presentation, text, audio, video. For definitions of the MMS Message Body structure refer to [MMSENC]. For a resource list body the content type as defined in [RFC4826] shall be used.

Table 65: CPIM Header of the Multimedia Message Object

Example of a recorded multimedia message:

```
From: tel:+4917112345678
To: Joe User <joe@user.org>, tel:0401234567
Cc: tel:+358501234567
Date: Tue, 27 01 2015 17:57:16 +0100
Subject: This is an example
Conversation-Id: 4465a75f-25f5-46dc-8da5-773a1ba174fc
Contribution-Id: f592acbb-fef9-489f-84be-e4596b607c6d
Message-Id: <12648104060@pf.operator.com>
Message-Correlator: 5478613765654386@mmsc.operator.com
IMDN-Message-ID: 654131a654131a654131a654131a8994123
Message-Direction: received
Message-Context: "multimedia-message"
Content-Type: Message/CPIM

From: tel:+4917112345678
To: Joe User <joe@user.org>
To: tel:0401234567
cc: tel:+358501234567
DateTime: 2015-01-27T16:32:55+01:00
Subject: This is an example
NS: imdn <urn:ietf:params:imdn>
NS: rcs <http://www.gsma.org>
imdn.Message-ID: 654131a654131a654131a654131a8994123
imdn.Disposition-Notification: display
rcs.Message-Correlator: 5478613765654386@mmsc.operator.com
rcs.Message-Context: "multimedia-message"
rcs.Mms-Message-Class: "Personal"

Content-Type: Multipart/Mixed;
              boundary="body-1"

     --body-1
   Content-Type: application/resource-lists+xml
   Content-Disposition: recipient-list-history; handling=optional

     <?xml version="1.0" encoding="UTF-8"?>
     <resource-lists xmlns="urn:ietf:params:xml:ns:resource-lists"
             xmlns:cp="urn:ietf:params:xml:ns:copycontrol">
     <list>
       <entry uri="mailto:joe@user.org" cp:copyControl="to" />
       <entry uri="tel:0401234567" cp:copyControl="to" />
       <entry uri="tel:+358501234567" cp:copyControl="cc" />
     </list>
     </resource-lists>

     --body-1
Content-Type: Multipart/Related;
              boundary="example-1";
              start=<950118>;
              type=application/smil

     --example-1
     Content-Type: application/smil
```

```
Content-ID: <950118>"

  [presentation]

--example-1
Content-Type: text/plain
Content-ID: <950119>

  [text]

--example-1
Content-Type: image/jpeg
Content-ID: <950120>

  [image]

--example-1—
--body-1--
```

4.1.4.7.2 Recording of MM1 Delivery Report

The entity storing a mobile originated MMS messages in the Common Message Store shall also store MM1 Delivery Reports.

A MM1 Delivery Report is matched on the originating side to the original sent message by use of MMS message-id value. In case of multiple recipients the address of the recipient from which the delivery report has been received needs to be taken into account.

The MM1 Delivery Indication shall be used by the entity storing MMS messages in the Common Message Store to record a Delivery Notification as defined in [RCS-CPM-CONVFUNC-ENDORS].

A delivery notification shall be recorded if the MMS Status field in the MMS delivery report indicates that the short message has been "retrieved" (see [MMSENC]). In all other cases a delivery notification shall not be recorded.

A MMS delivery notification shall be recorded in the Common Message Store with mail and CPIM headers set as follows.

Mail Header	Status	Content
From	Mandatory	It contains the address of the recipient of the original message as derived from the MMS deliver indication, see [MMSENC]. If identified by a E.164 number it is encoded in the global-number representation of a tel URI as defined in [RFC3966] If identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. If identified by an e-mail address it is encoded as mailbox as defined in [RFC5322].

Mail Header	Status	Content
To	Mandatory	It contains the public user identity of the sender of the original message. It is encoded in the global-number representation of a tel URI as defined in [RFC3966] or as a SIP URI as defined in [RFC3261].
Date	Mandatory	Indicates the time the message was recorded
Conversation-ID	Mandatory	It shall have the same value as the original sent message.
Contribution-ID	Mandatory	It shall have the same value as the original sent message.
IMDN-Message-ID	Mandatory	It shall be assigned by the entity that stores the message. The IMDN-Message-ID header value should generated with similar definition and uniqueness requirements as described in section 6.3 of [RFC5438]. In a transition period Service Providers may store in the IMDN-Message-ID header the value of the MMS Message-ID for compatibility with [RCS-CPM-MSGSTOR-ENDORS].
Message-ID	Optional	It is assigned by the entity that stores the message. If stored, the header value should conform to the definitions of [RFC5322]. In a transition period Service Providers may store in the Message-ID header the value of the MMS Message-ID for compatibility with earlier versions of this specification.
Message-Correlator	Mandatory	It shall contain the MMS Message-ID for message correlation as defined in section 4.1.4.3. See also section 4.1.4.2. It has the same value as the Message-Correlator header of the original sent message.
Content-Type	Mandatory	Message/CPIM

Table 66: Mail Headers of the MMS Delivery Notification

The Message Body of the MMS Delivery Notification shall contain the following CPIM headers:

CPIM Header	Status	Content
From	Mandatory	It contains the address of the recipient of the original message as derived from the MMS deliver indication, see [MMSENC]. If identified by a E.164 number it is encoded in the global-number representation of a tel URI as defined in [RFC3966] If identified by a non E.164 number then it is encoded in the local-number representation of a tel URI as defined in [RFC3966] without parameters and context. If identified by an e-mail address it is encoded as mailbox as defined in [RFC5322].
To	Mandatory	It contains the public user identity of the sender. It is encoded in the global-number representation of a tel URI as defined in [RFC3966]
DateTime	Mandatory	It should indicate the time the message was delivered by the Service Centre. The value is derived from the Date field of the MM1 Delivery report, see [MMSENC]. If it is not available then it should contain the time of message recording.

CPIM Header	Status	Content
imdn.Message-ID	Mandatory	Shall be set to the same value as the corresponding MIME header (see Table 66).
rcs.Message-Correlator	Mandatory	It shall contain the MMS Message-ID for message correlation as defined in section 4.1.4.3. See also section 4.1.4.2. It has the same value as the rcs.Message-Correlator header of the original sent message.
Content-Type	Optional	message/imdn+xml

Table 67: CPIM Header of the MMS Delivery Notification

The message body of the CPIM message shall contain an IMDN with status set to "delivered" and the imdn.Message-ID assigned to the original sent message.

4.1.4.7.3 Recording of MMS Read Reports

According to [MMSCTR] there are two methods in MMS to transfer a Read Report from the recipient to the sender.

1. Multimedia Message Read Report:
   After display of a MMS message for which the originator has requested a read reply and the sending of a read report is authorised by the recipient user the client generates an "automatic" MMS message and sends it back to the sender. The entity storing MMS messages in the Common Message Store shall store the multimedia message read report message as a normal MMS message as defined in section 4.1.4.7.1. It is essential that the storing entity sets the value of the rcs.Mms-Message-Class header with the value received in the message, i.e. "auto".
   Clients fetching a MMS message and the related multimedia message read report from the Common Message Store should present it in the message history in the same way as if the message and the read report would have been received via MMS.
2. PDU Read Report:
   The implementation of recording of MMS PDU Read Reports is for further study.

A client sending a MMS Read Report to the sender shall not regard it as a display IMDN notification which is used by the messaging server participating functions as a trigger to set the \Seen flag in the Common Message Store. Therefore the client shall follow the procedures for the setting of the \Seen flag in Common Message store defined in section 4.1.4.8 as if an IMDN display notification was not requested.

4.1.4.8 Optimisations for UNI operations to Common Message Store

If the MESSAGE STORE EVENT REPORTING configuration parameter is enabled (see section A.1.4.3), the RCS client shall use the CPM event reporting framework procedures (see section 6.7 of [RCS-CPM-CONVFUNC-ENDORS]) in an established 1-to-1 Chat session.

The RCS client shall use this for the following cases between RCS client and Participating Function, in order to report:

1. when one or more messages have been read by the RCS user, so that the Participating Function can set the “\Seen” flag for the stored message object(s) in CMS on behalf of the RCS client; and,
2. When a message was deleted by the RCS user, so that the Participating Function can set the “\Deleted” flag for the stored message object(s) in CMS.

To set the “\Seen” flag (case 1):

When the RCS user has read/displayed a received message, the RCS client shall inform the Common Message Store so that the message will be shown as “read” on other user devices after synchronizing with the Common Message Store.

This can be realised in the following ways:

1. If an IMDN display notification was requested for the received message and one was generated by the RCS client, the Participating Function (B2BUA) receiving the IMDN display notification from the client (via MSRP or SIP MESSAGE request) shall set the “\Seen” flag in CMS for the messages reported;
2. If an IMDN display notification was not requested, or if the RCS user settings on the client disabled sending them for read messages, then the CPM event reporting framework is used to report to the Participating Function that a message was read by the RCS client.

To set the “\Deleted” flag (case 2):

When the RCS user has deleted a received message, the RCS client shall inform the Common Message Store so that the message will be shown as “deleted” on other user devices after synchronizing with the Common Message Store. This is realised by the CPM event reporting framework being used to report to the Participating Function that a message was deleted by the RCS client.

4.1.4.9 A Common File Store for File Transfer via HTTP

4.1.4.9.1 Overview

For the File Transfer via HTTP option, storage for files/content transferred via HTTP is kept separate from the smaller, text based content and metadata. It also may be desirable to keep received content for longer than the original validity period. For these purposes, a Common File Store may be used together with a Common Message Store. The Common Message Store stores the chat messages carrying the XML containing the information for the thumbnail (optional) and the file(content), while the Common File Store stores the local copy of the thumbnail and/or content indicated by those URL links.

Similar to the Common Message Store, the Common File Store may be used to synchronise files between devices in the recipient’s network in addition to keeping a back-up of files in the local network.

The Common File Store is meant both for sent and received files and requires the existence of a Common Message Store. When the Common File Store is deployed, it shall behave as the Content Server for that network supporting File Transfer via HTTP procedures for both senders of files in that network (see section 3.5.4.8.3.1) and receivers of files in that network (see section 3.5.4.8.3.2).

For the Common File Store to work with the Common Message Store a function named File Transfer Localisation function is required. The File Transfer Localisation function shall always be triggered upon a client/device request to download a file. This is achieved through the Messaging server inserting a prefix or overwriting the HTTP URL for the thumbnail (if provided) and the HTTP URL for the file with an HTTP URL pointing to the File Transfer Localisation Function. As for the values of the other XML parameters (e.g. until = "[validity of the file]"), they are unchanged and remain relevant.

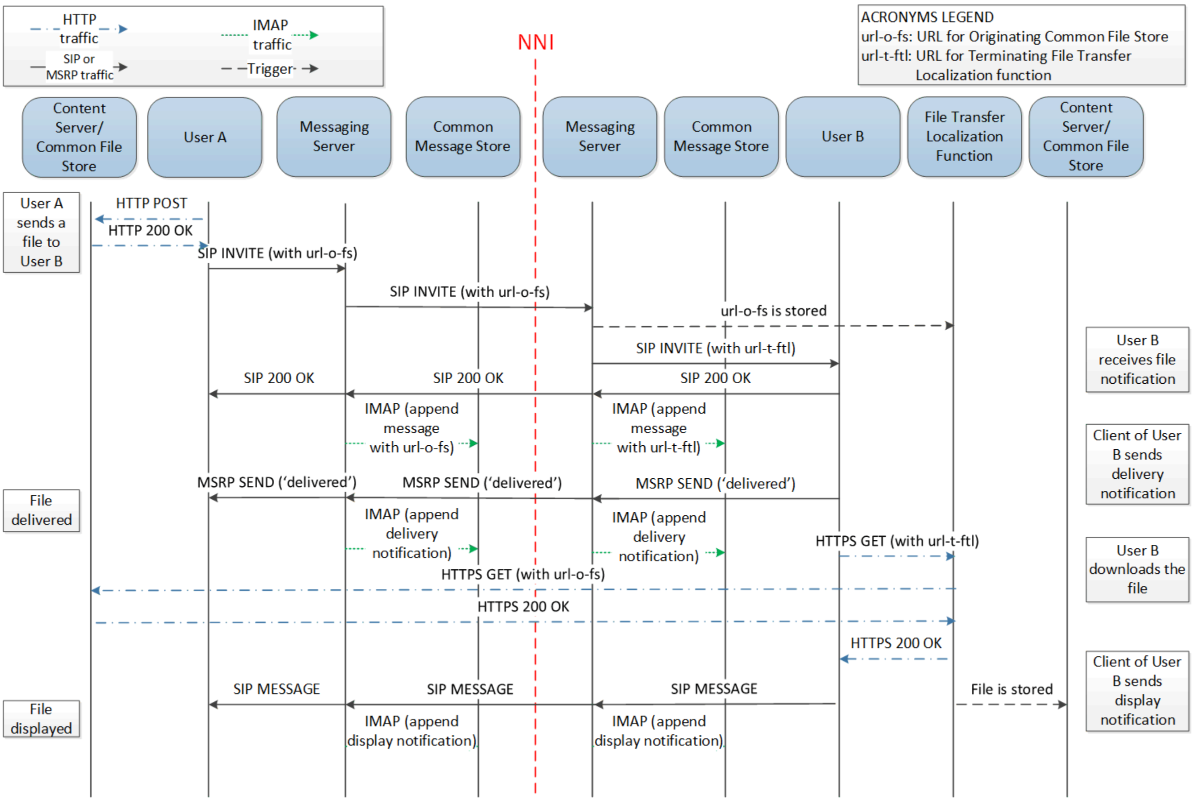

Figure 119: File Transfer Localisation Function: global flow

4.1.4.9.2 Sender Procedures

The Service Provider shall direct the files uploaded by the sender towards a permanent storage with or without triggering the File Transfer Localisation function. The same or a different location can be used for uploading files sent by legacy clients.

The procedures as described in section 3.5.4.8.3.1 apply. Please note that referring to the XML body in Table 37 the following should be taken into account:

- The validity indicates how long the file is available to the receiver for first time file download request. In the case where the Common File Store is deployed with the Common Message Store, for the XML in sent File Transfer via HTTP messages retrieved from the Message Store Server the validity in the XML is not relevant and should be ignored since the validity period configured for sent files in the Common File Store applies.

NOTE: When a Common File Store is deployed, it is up to service provider policy whether the sent files are assumed to be archived if the Archived flag is set for the link to the file.

Additionally, the upload resume procedures described in 3.5.4.8.3.1.1 remain relevant.

In the case where the Common File Store is deployed for file/content synchronisation purposes for other devices of the user the retrieval procedures defined in section 4.1.4.9.4 apply.

4.1.4.9.3 Terminating Messaging Server Procedures

As described in section 4.1.4.9.1 a receiver's request to download a file shall always trigger the File Transfer Localization function if the Common File Store is also deployed. The HTTP URL for the thumbnail (optional) and the HTTP URL for the file included in the XML received by the client shall direct the HTTP GET request towards the File Transfer Localisation function.

In order to ensure such a re-direct, the Messaging Server shall prefix or overwrite any URL contained in the *url* attribute of the data element of a File transfer via HTTP message body content for both the thumbnail (if included) and the file itself.

4.1.4.9.4 Receiver Procedures

The receiver procedures are unchanged compared to section 3.5.4.8.3.2 when receiving the File Transfer request from another user. In particular this means that any response received from the remote side is proxied through the File Transfer Localization Function back to the RCS client.

Note that the validity in the XML received by the client shall indicate to the client how long the file is available for first time download request. Once the client sends the file download request, the file is localized and made available for future retrievals. In the case where the Common File Store is deployed with the Common Message Store, for the XML in File Transfer via HTTP messages retrieved from the Message Store Server, the validity in the XML is not relevant and should be ignored since the validity period configured for localized files in the Common File Store applies.

NOTE1: When a Common File Store is deployed, it is up to service provider policy whether the received files are assumed to be archived if the Archived flag is set for the link to the file.

Once the user sends the HTTP GET request for first download or subsequent file retrieval, a similar authentication procedure as described in section 3.5.4.8.3.1 should be initiated. The HTTP state management recommendations in section 3.5.4.8.7 also apply. Specifically, the HTTP GET request towards the Localisation Function should result in a HTTP 401 AUTHENTICATION REQUIRED error response carrying a WWW-Authenticate header field as defined in [RFC2616]. The receiver shall in this case depending on the WWW-Authenticate header received then make another HTTP GET request using basic authentication or HTTP digest as per [RFC2617] using *FT HTTP CS USER* and *FT HTTP CS PWD* configuration parameters as credentials (defined in Table 76 in section A.1.5) or HTTP digest using a bootstrapped security association as per [3GPP TS 24.109].

NOTE2: It is left to service provider policies to ensure that the authentication procedure is not triggered towards legacy clients.

4.1.4.9.5 File Transfer Localization function procedures

The File Transfer Localisation function, once triggered, shall either download the file from the location that the file sender has uploaded it (the file sender's Content Server or Common File Store for first time download) or retrieve it from the Common File Store (for any subsequent download). For the former case, the File Transfer Localisation function shall also upload the file to the Common File Store so it is available for any subsequent download requests coming from a different device of the same user.

The File Transfer Localisation function shall keep track for every file transfer of the HTTP URL provided by the sender so it can download the file upon first client download request. Whether or not the File Transfer Localisation function is triggered for files sent to a legacy device is left up to the Service Provider implementation.

4.1.5 NNI and IOT considerations

Since Messages are only stored after delivery, the Message Store has no impact on the NNI.

4.1.6 Implementation guidelines and examples

Clients shall comply with the operations and procedures described in sections 5.5.2 of [CPM-SYS_DESC] and 6 of [RCS-CPM-MSGSTOR-ENDORS]. Some further clarifications on the client expected behaviour when it interacts with the Message Store Server are presented in the following sections.

4.1.6.1 Authentication and Transport Confidentiality

Before sending any IMAP commands, the client shall set up a TLS connection. The client shall be configured to support only strong encrypted sessions by default. In case the client's supported strong ciphers are downgraded, the client shall terminate the IMAP connection and should attempt to re-establish the TLS session.

The client shall send a CAPABILITY command once a TLS connection has been set up (as per [RCS-CPM-MSGSTOR-ENDORS]). It shall also send a CAPABILITY command once authentication is completed. Client required behaviour upon authentication is reflected in Annex B.4.

As per [RFC3501], a Common Message Storage server may send capabilities automatically as part of a response to a previous command. In that case, the client should not send a separate CAPABILITY command.

4.1.6.2 Storing new messages (Object Store Operation)

For the cases that the client stores new messages and there is no existing folder where these messages can be stored (e.g. a brand new conversation with user B offline and interworking procedures in place), the client needs to allocate the name to the new folder and follow the naming procedures as described in 6.3.1 of [RCS-CPM-MSGSTOR-ENDORS].

4.1.6.3 Message Archive

When a user wants to archive a message, the client shall set the Archived flag on that message.

When a user wants to unarchive a message the client shall remove the Archived flag from that message.

For the case that permanent message storage is required due to Service Provider policy, the client is configured with the MESSAGE STORE ARCHIVE AUTH set to disabled and the client does not need to set the Archived flag.

4.1.6.4 Search Operation

Search operation shall first be performed locally when local storage exists. A search may be expanded to a network based search.

4.1.6.5 Message Displayed on a Client (Object Store Operation to set a flag)

When a client displays a previously unseen message for the user then it shall trigger the change of the message flag in the Common Message Store as follows:

If a message

- was originally received at a time when there the Common Message Store was permanently disabled for the client and it has not been uploaded (e.g. by user activity) or
- the message is kept only locally stored, e.g. after it was deleted from the Default folder as defined in section 6.7 of [RCS-CPM-MSGSTOR-ENDORS]

then the client shall manage the state only locally.

In all other cases the client shall attempt to update the message status in the Common Message Store with the procedures defined below.

If the message status is changed to seen and the client does not send a display notification then the client shall act depending on the value of the MESSAGE STORE EVENT REPORTING configuration parameter (see section A.1.4.3) as follows:

1. If the MESSAGE STORE EVENT REPORTING configuration parameter is disabled, the client shall set the \Seen flag for that message right after it displays the message as follows.

    - If the UID of the message for which the flag is to be set is not known because it was previously sent or received but not synchronised with the Common Message Store yet, the client shall first attempt to match it with the message in the Common Message Store as defined for synchronisation in section 4.1.6.8. If no match is found the message is handled as defined in section 4.1.6.8, otherwise it shall follow the steps below.
    - If there is no connection with the IMAP server, the client shall establish a connection to it.

- If there is a connection to the IMAP server then the client shall use it to set the flag. If the session with the IMAP server is currently used for synchronisation as defined in section 4.1.6.8 the client shall

  - stop the synchronisation procedure at the next reasonable point (e.g. after completion of the ongoing IMAP command or synchronisation action),
  - perform the procedure to change the flag,
  - continue the synchronisation from the point where it was stopped.

- Then the client shall select the conversation folder where the message is stored on the IMAP server.
- When setting the flag, the client shall follow procedures as described in section 6.6 of [RCS-CPM-MSGSTOR-ENDORS].
- If the IMAP server returns a success response the client shall inspect the updated flags in the response. If the result in the response is different than the local message status, then the client shall synchronise messages in the conversation folder in accordance with the definitions in section 4.1.6.8.
- If the IMAP server returns an error response the client shall synchronise messages in the conversation folder in accordance with the definitions in section 4.1.6.8.

Optimisations for setting the \Seen flags when many messages are displayed or deleted by the client in a short period of time are left to the device implementation, e.g. by keeping the IMAP connection open as long the user has the messaging application opened or until a client local inactivity timer expires.
If the connection with the IMAP server has been established by the client specifically to manage message status but not by a synchronisation trigger defined in section 4.1.6.8 the client may use this session to run a full synchronisation as defined in section 4.1.6.8.

2. If the MESSAGE STORE EVENT REPORTING configuration parameter is enabled, the client shall inform the Messaging Server as per section 4.1.4.8 so that the Messaging Server will set the \Seen flag.

For client fetching flags, procedures as described in section 6.6 of [[RCS-CPM-MSGSTOR-ENDORS] apply.

4.1.6.6 Message Removal via Clients (Object Store Operation to set a flag and possibly Object Remove Operation)

When the client deletes a message from the Default folder from one device, all other clients belonging to the user shall also show that message as deleted.

When the client deletes a message which

- was originally received at a time when there the Common Message Store was permanently disabled for the client and it has not been uploaded (e.g. by user activity) or
- is kept only locally stored, e.g. after it was deleted from the Default system folder as defined in section 6.7 of [RCS-CPM-MSGSTOR-ENDORS]

then the deletion is managed only locally.

In all other cases the client shall delete the message in the Common Message Store with the procedures defined below.

A client deletes a message from the Default folder in one of two ways, depending on the value of the MESSAGE STORE EVENT REPORTING configuration parameter (See section A.1.4.3):

1. If the MESSAGE STORE EVENT REPORTING configuration parameter is disabled, the client shall set the \Deleted flag for that message right after the local deletion. The client shall set the flag in a way that the \Deleted flag is set and the \Archived flag is removed if it had been set, and all other flags remain set.

   - If the UID of the message for which the flag is to be set is not known because it was previously sent or received but not synchronized with the Common Message Store yet, the client shall first attempt to match it with the message in the Common Message Store as defined for synchronization in section 4.1.6.8. If no match is found the message is handled as defined in section 4.1.6.8, otherwise the client shall follow the steps below.
   - If there is no connection with the IMAP server, the client shall establish a connection to it.
   - If there is a connection with the IMAP server the client shall use it to set the flag. If the session is currently used for synchronization as defined in section 4.1.6.8 the client shall

     - stop the synchronisation procedure at the next reasonable point (e.g. after completion of the ongoing IMAP command or synchronisation action),
     - perform the procedure to change the flag,
     - continue the synchronisation from the point where it was stopped.

   - The client shall select the conversation folder in the Default folder where the message is stored on the IMAP server.
   - When setting the flag, procedures as described in section 6.6 of [RCS-CPM-MSGSTOR-ENDORS] apply.
   - If the IMAP server returns success response the client shall inspect the updated flags in the response. If the result in the response is different than the local message status, then the client shall synchronise messages in the conversation folder in accordance with the definitions in section 4.1.6.8.
   - If the IMAP server returns an error response the client shall synchronise messages in the conversation folder in accordance with the definitions in section 4.1.6.8.

   Optimisations for setting the \Deleted flags when many messages are being displayed or deleted by the client in a short period of time are left to the device implementation, e.g. by keeping the IMAP connection open as long the user has the messaging application opened or until a client local inactivity timer expires
   If the connection with the IMAP server has been established by the client specifically to manage message status but not by a synchronisation trigger defined in section

4.1.6.8 the client may use this session to run a full synchronisation as defined in section 4.1.6.8.

2. If the MESSAGE STORE EVENT REPORTING configuration parameter is enabled, the client shall inform the Messaging Server as per section 4.1.4.8 so that the Messaging Server will set the \Deleted flag.

4.1.6.7 Client impact of Message Removal due to Service Provider Policy

When the server deletes (expunges) messages in the Default folder because of e.g. message expiry, the client Shall Not remove these same messages from their own device.

4.1.6.8 Synchronization

NOTE: Client synchronization guidelines are built under the assumption that roaming scenarios are out of scope. Synchronisation triggers may change under roaming scenarios

The following types of triggers for synchronisation apply:

1. Data Connection State Triggered Synchronisation
   This type of synchronisation trigger includes all the cases where the device moves from disconnected to connected state. This includes the following triggers:

   a) the device is powered on or the client is launched
   b) regain of data connection due to toggle of the operating system's "data on/off switches", e.g. "data traffic switches" or "flight mode"
   c) in the case of other data connection state triggers and if the client detects a loss of data connection when setting up or during a connection with the message store or during other network interactions it should (re-) connect to the message store on connectivity re-gain (see also section 2.4.7.6).

   All folders with new content and/or flags are expected to be synchronised and no folder prioritisation for synchronisation applies. Connection state triggered synchronisation happens in the client background and once triggered, the client shall initiate the login and first time synchronisation procedures (see use cases 1 and 2 in section B.4). Once the synchronisation procedure is completed, client is expected to logout. Clients with bearer control capabilities should attempt to logout from an existing IMAP session prior to data connectivity loss.
   Once a data connection state related synchronisation trigger is sent towards the Common Message Storage server, the client shall ignore any subsequent data connection state synchronisation triggers unless the DATA CONNECTION SYNC TIMER has expired. This parameter is described in section A.1.4.

2. Periodic synchronisation

   Periodic synchronisation shall be triggered within a configurable time interval set by the Service Provider (please refer to MESSAGE STORE SYNC TIMER configuration parameter as described in Table 75). As for synchronisation due to data connectivity re-gain, all folders are expected to be synchronised and no folder prioritisation for synchronisation applies. Periodic synchronisation happens in the client background and once triggered, the client shall initiate the login and first time synchronisation procedures (see use cases 1 and 2 in section B.4). Once the synchronisation procedure is completed, the client should logout.

3. Synchronisation triggered by User activity

   This type of synchronisation shall be triggered once the user enters a particular conversation. For this type of synchronisation, all folders are considered for the synchronisation procedure but prioritisation is given to the conversation that the user enters right before synchronisation triggering. The synchronisation of the rest of the conversations happens right after the synchronisation of the prioritised conversation is completed and it takes place in the background.
   Any synchronisation triggered by user activity prior to the completion of ongoing synchronisation shall be ignored by the client and consequently it shall not be propagated towards the Common Message Storage server. The synchronisation is considered completed once content changes have been updated in all conversations.
   To reduce the number of session establishments while the user manages messages in the conversation history, the client should apply a smart handling of IMAP connections for this type of synchronisation, e.g. to keep the connection open as long the user has the messaging application opened or until a client local inactivity timer expires.

When a RCS client is triggered for synchronisation it shall follow the procedures defined in section 6.7 of [RCS-CPM-MSGSTOR-ENDORS]. Specifically it shall consider the folder structure as defined in Object Store Operation described in 6.3.1 of [RCS-CPM-MSGSTOR-ENDORS].

To find new messages, messages with updated or new flags and expunged messages the client shall use “CONDSTORE”/“QRESYNC” as described in accordance with section 6.7 in [RCS-CPM-MSGSTOR-ENDORS] and shall use “XLIST-CHANGEDSINCE” when supported by the server. The server shall support “CONDSTORE”, “QRESYNC” and “XLIST-CHANGEDSINCE” as per section 6 in [RCS-CPM-MSGSTOR-ENDORS].

If triggered for synchronisation the client shall match all new sent or received messages and disposition notifications with messages of the Common Message Store as follows:

If

- a new message has been sent via SMS or MMS or
- a new message has been received via SMS or MMS or
- a delivery report has been received via SMS or MMS or
- a read report has been received or MMS or
- a flag has been changed on a message object

the client shall follow the procedures for matching of messages defined in sections 4.1.4.2.1 and 4.1.4.3.

If

- a message has been sent or

- a message has been received or
- a delivery notification has been received or
- a display notification has been received or
- a flag has been changed on a message object

via standalone messaging, chat, group chat and file transfer, the client shall follow the procedures for matching of messages as defined for these services in [RCS-CPM-MSGSTOR-ENDORS].

For standalone messages with Pager Mode, it is possible based on Service Provider policies, to be automatically stored in the CMS as legacy SMS messages. In this case the client will not be able to correlate these messages with their copy in the Common Message Store based on the procedures for matching of messages defined in [RCS-CPM-CONVFUNC-ENDORS]. If the client is not able to correlate the standalone messages with Pager mode based on the procedures defined in [RCS-CPM-CONVFUNC-ENDORS] it shall then attempt to correlate them based on the procedures defined for SMS messages in sections 4.1.4.2, 4.1.4.3, 4.1.4.4 and 4.1.4.5.

When a client attempts to match a new sent or received message with the Common Message Store and no match is found, then

- If the message was received via SMS or MMS and the corresponding client configuration parameter SMS MESSAGE STORE or MMS MESSAGE STORE defined in Table 75 is set to value "1", then the client shall apply the procedure defined in section 4.1.4.2.1.
- In all other cases the client shall keep the message locally stored. The client shall try to match the message again at least in the next synchronisation. The client shall perform message management for this message only locally.

NOTE: There may be cases where the client will not find a match because the message has been deleted and expunged in the Common Message Store by user activity from another client. However there is no procedure for the client to detect this scenario. In this case the user may need to delete the same message again manually on this client.

For a new sent or received message, if a match is found, then the client shall align the local message status with the status in the Common Message Store as defined in Table 68.

		Message Status on Server			
		not flagged	**seen**	**deleted**	**archived**
Message Status on Client	**not flagged**	no action	set local message seen	delete local message	set local message archived
	seen	set \Seen flag on server message	no action	delete local message	Set \Seen flag on server message (most likely already set)
	deleted	set \Deleted flag on server message	set \Deleted flag on server message	no action	set \Deleted flag on server message
	archived	set Archived flag on server message	set Archived flag on server message	delete local message	no action

Table 68: Message Status Synchronisation for new Messages

The procedures for the client for setting a message seen in the Message Store Server are described in section 4.1.6.5.

The procedures for the client to delete messages in the Message Store Server are described in section 4.1.6.6.

The client should consider the size of messages to be fetched in order to avoid delaying the fetch of the body of basic messages (chat messages, standalone messages and SMS/MMS messages). The fetch of the body of larger messages may come later in the synchronisation procedure.

When an RCS client fetches a new object, it shall fetch it in a way so that it remains unread/unseen if it had not yet been read/seen by the user (i.e. use of BODY.PEEK).

If a message was originally received at a time when the Common Message Store was permanently disabled or a message is kept only locally stored, e.g. after it was deleted from the "Fefault" system folder as defined in section 6.7 of [RCS-CPM-MSGSTOR-ENDORS], then the client should not consider it for synchronisation. However the user or client may consider to upload such messages to the Default folder.

Upon synchronisation, clients shall find all messages that might have been deleted on the Common Message Store by other devices. The client shall delete the local message if a message in a conversation folder in the Default folder is flagged \Deleted.

If a message in a conversation folder of the default folder is expunged, the client shall continue to keep the message locally.

4.1.6.9 Identification of Message Direction of Messages in a conversation

To simplify the client procedure to display messages as sent or received within a conversation an entity storing the message in the Common Message Store shall add the message transfer direction via a Message-Direction mail header.

The Message-Direction mail header should be stored for all types of user messages in the Common Message Store.

If the Message-Direction header is present and the value is set to "sent", then the client shall display the message as sent from the own user. The client should disregard the address value in the From: header of the message. If the Message-Direction header is present and the value is set to "received", then the client shall display the message as received by the own user. The client should disregard the address value in the To: header of the message.

If the Message-Direction header is not present then the client should display the message direction based on a local device policy.

The Message-Direction header is defined in section C.3.1.

Annex A Managed objects and configuration parameters

This Annex provides the full details on the RCS data model including an overview of all configuration parameters. These parameters will be set using the mechanisms described in section 2.3.

The aim of this section is to provide a complete configuration data model for reference by both Service Providers and OEMs.

A.1. Management objects parameters overview

This section provides an overview of the configuration parameters used for RCS. These parameters can either come from an existing management object (like for instance the OMA defined objects for Presence, Messaging and so on) or may be RCS specific. In the latter case they will be formally defined in section A.2.

NOTE: This may not be the only document where parameters controlling an RCS device are defined (see e.g. [PRD-RCC.53] and [PRD-RCC.15]).

A.1.1. RCS status related configuration

This RCS specification includes the following RCS Specific configuration parameters controlling the status of the RCS client on the device:

Configuration parameter	Description	RCS usage
RCS DISABLED STATE	Controls the way in which an RCS client is disabled on the device. It can have following values resulting in the behaviour described in section 2.3.3.2.1: • 0, the client is temporarily disabled • -1, the client is permanently disabled • -2, the client is permanently disabled, but user triggered events result in an attempt to re-enable the client • -3, the client is placed in dormant state If included a configuration document should not include other RCS related configuration parameters. Note: if not included, the RCS client should be enabled and the document should include a valid configuration for the other RCS configuration parameters.	Optional Parameter

Table 69: RCS status configuration parameters

A.1.2. Presence related configuration

A.1.2.1. OMA Presence Provisioning parameters

OMA Presence Client provisioning parameters are defined in [PRESENCE2MO]. Table 70 lists the OMA Presence parameters applicable to RCS.

Configuration parameter	Description	RCS usage
CLIENT-OBJ-DATA-LIMIT	maximum size of the MIME object in SIP PUBLISH request	Optional parameter It is mandatory and becomes relevant only if DEFAULT DISCOVERY MECHANISM is set to PRESENCE or PRESENCE PROFILE is set to 1. In that case it will be set to a fixed value of 524288 (i.e. 512KB).
CONTENT-SERVER-URI	HTTP URI of the content server to be used for content indirection	Not Used
SOURCE-THROTTLE-PUBLISH	minimum time interval (in seconds) between two consecutive publications	Optional parameter It is mandatory and becomes relevant only if DEFAULT DISCOVERY MECHANISM is set to PRESENCE or PRESENCE PROFILE is set to 1
MAX-NUMBER-OF-SUBSCRIPTIONS-IN-PRESENCE-LIST	Limits the number of back-end subscriptions allowed for a presence list. This parameter applies to the “rcs” list (as described in section 3.7.4.5)	Optional parameter It is mandatory and becomes relevant only if PRESENCE PROFILE is set to 1

SERVICE-URI-TEMPLATE	syntax of the service URI	Optional parameter It is not provided for RCS which uses a value of “<xui>;pres-list=<id>” according to section 5.5.1 in [PRESENCEIG]
RLS-URI	SIP URI of the RLS to be used by the Watcher when subscribing to a Request-contained Presence List Default if not provided: *presence-rls@ims.mnc<MNC>.mcc<MCC>.pub.3gppnetwork.org* whereby <MNC> and <MCC> shall be replaced by the respective values of the home network in decimal format and with a 2-digit MNC padded out to 3 digits by inserting a 0 at the beginning	Optional parameter

Table 70: RCS usage of OMA presence configuration parameters

A.1.2.2. RCS Specific Provisioning parameters

This RCS specification includes the following additional presence related configuration parameters:

Configuration parameter	Description	RCS usage
PRESENCE PROFILE	This parameter allows enabling or disabling the usage of the social information via presence. If set to 0, the usage of the social information via presence feature is disabled. If set to 1, the social information via presence feature is enabled. This parameter will consequently influence the inclusion of the tag associated with social information via presence in OPTIONS exchanges.	Mandatory parameter
NOTE MAX SIZE	Maximum length of presence tagline at presentity side. The reason to have at presentity side a configurable attribute on the RCS client to control the maximum size of the Note is to make the end user aware of what the limit is (when typing the content of the Note/free text). Avoiding enforcement of this limit at network / watcher side would lead to truncating the note. This value should have a lower value than the one defined at watcher side in the OMA Presence Implementation guideline [PRESENCEIG].	Optional parameter It is mandatory and becomes relevant only if PRESENCE PROFILE is set to 1.

LOCATION VALIDITY	This parameter allows controlling the maximum time during which a location information should be considered valid.	Optional parameter It is mandatory and becomes relevant only if PRESENCE PROFILE is set to 1.
NON-VIP CONTACTS POLL MAX FREQUENCY	This parameter controls the maximum number of poll operations on the non-VIP contacts list during a certain period of time.	Optional parameter It is mandatory and becomes relevant only if PRESENCE PROFILE is set to 1.

Table 71: RCS additional presence related configuration parameters

A.1.3. XDM related configuration

A.1.3.1. OMA XDM Provisioning parameters

OMA XDM Client provisioning parameters are defined in [XDMMO]. The following table lists the OMA XDM parameters applicable to RCS. The whole tree becomes optional if no functionality depending on XDM is deployed (that is no Presence based capability check as described in section 2.6.1.2 and no Social Presence as described in section 3.7 or PS Voice or Video calls as described in section 3.8 and 3.9), or if the device is VoLTE or VoWiFi enabled. VoLTE or VoWiFi devices would use the default XCAP Root URI value as defined in [PRD-IR.92] or [PRD-IR.51] respectively, but the default value could still be overwritten with the OMA XDM parameter.

Configuration parameter	Description	RCS usage
XCAP Root URI	The root of all XCAP (XML configuration access protocol) resources (which points to the Aggregation Proxy address). This is used when accessing via XCAP. If not provided, this shall be handled as described in section 2.7.1 of [PRD-NG.102], also for devices that are not enabled for VoLTE/VoWiFi. If the AP needs to be accessible via direct WiFi, the value set for the XCAP Root URI needs to be routable in the public domain (as example, see the section 2.3.4 in [PRD-IR.67]).	Optional parameter

XCAP Authentication user name	HTTP digest "username", for accessing an XDMS (XDM server) using the XCAP protocol. The "username" shall be set to the IMPU of the RCS user, i.e. a SIP URI or a TEL URI, according to the procedures in section 5.1.1 and 5.1.2.1 of [XDM2.2_Core] and section 2.13.1.4 of this document.	Optional parameter It is mandatory and becomes relevant only if XCAP Authentication Type is set to "Digest".
XCAP Authentication Secret	HTTP digest password	Optional parameter It is mandatory and becomes relevant only if XCAP Authentication Type is set to "Digest".
XCAP Authentication type	Authentication method for XDMS over XCAP (see section 2.13.1.4). Possible values: GAA or Digest. If not provided, the default depends on the client type (see section 2.2 and 2.13.1.4): • Embedded RCS clients shall use GAA • Any other RCS clients (e.g. Downloadable clients and clients on secondary devices) shall use Digest	Optional parameter

Table 72: RCS usage of OMA XDM configuration parameters

A.1.3.2. RCS Specific Provisioning parameters

This RCS specification includes the following additional XDM related configuration parameters:

Configuration parameter	Description	RCS usage
REVOKE TIMER	This parameter allows setting the duration during which a contact should remain in the "rcs_revokedcontacts" list (as described in section 3.7.4.5). It may also be used for the frequency that the list is checked.	Optional parameter It is mandatory and becomes relevant only if PRESENCE PROFILE is set to 1.
PNB MANAGEMENT	This parameter allows to enable (1) or disable (0) the PNB feature (as described in section 2.15.1)	Optional parameter. If not present, it is assumed that PNB feature is disabled.

Table 73: RCS additional XDM related configuration parameters

A.1.4. Chat related configuration

A.1.4.1. OMA SIMPLE IM Provisioning parameters

OMA SIMPLE IM client provisioning parameters are defined in [RCS-SIMPLEIM-ENDORS]. Following table only lists which of those SIMPLE IM application parameters are applicable.

Configuration parameter	Description	RCS usage
PRES-SRV-CAP	Flag used for the Messaging Server to indicate the Presence publish capability of a Presence information element of the Messaging Server on behalf of the SIMPLE IM Client	Not Used. Always set to the OMA value indicating that the capability is not supported in the network
MAX_AD-HOC_GROUP_SIZE	Maximum number of Participants allowed for an Ad-hoc Group Chat session	Optional parameter It is mandatory and becomes relevant only if GROUP CHAT AUTH is set to 1 or is not provided and CHAT AUTH is set to 1
CONF-FCTY-URI	SIP URI used for setting up an Ad-hoc Group or extending a 1-1 Chat session. Default value if not provided: chat@conf-factory.<home-network-domain-name> Where <home-network-domain-name> is replaced with the Home Network Domain Name used by the client (see [PRD-RCC.15] and [PRD-NG.102])	Optional parameter It is mandatory and becomes relevant only if Group Chat is enabled.
EXPLODER-URI	SIP URI used for sending SIP MESSAGE (e.g. Sending SIP MESSAGE to an Ad hoc Group) Default value if not provided: exploder@conf-factory.<home-network-domain-name> Where <home-network-domain-name> is replaced with the Home Network Domain Name used by the client (see [PRD-RCC.15] and [PRD-NG.102])	Optional parameter
CONV-HIST-FUNC-URI	SIP URI for the SIMPLE IM user’s conversation history storage	Not Used, populated with “sip:foo@bar”
DEFERRED-MSG-FUNC-URI / MSG-STORE-URI	SIP URI used for the SIMPLE IM User’s message-store account for deferred messaging	Not Used, populated with “sip:foo@bar”

Table 74: RCS usage of OMA SIMPLE IM configuration parameters

A.1.4.2. OMA CPM Provisioning parameters

OMA CPM does not include any formal provisioning parameter definition. Therefore the parameters for CPM are defined as RCS specific in section A.1.4.3. Furthermore following SIMPLE IM Parameters (see section A.1.4.1) will be applicable also for CPM services:

- MAX_AD-HOC_GROUP_SIZE
- CONF-FCTY-URI
- EXPLODER-URI
- DEFERRED-MSG-FUNC-URI / MSG-STORE-URI

NOTE: If standalone messaging is enabled (see section A.1.4.3), this parameter can be set to a value different from sip:foo@bar in which case 1-to-Many standalone messaging can be used.

A.1.4.3. RCS Specific Provisioning parameters

This RCS specification includes the following additional Chat related configuration parameters:

Configuration parameter	Description	RCS usage
CHAT AUTH	This parameter Enables/Disables the Chat service. If set to **0** the Chat service is disabled. When set to **1** it is enabled.	Mandatory Parameter
GROUP CHAT AUTH	This parameter Enables/Disables the Group Chat service. If set to **0** the Group Chat service is disabled. When set to **1** it is enabled. If not present, the CHAT AUTH parameter is used to determine Group Chat enablement. If CHAT AUTH is disabled (0), GROUP CHAT AUTH shall be also disabled (0) if configured.	Optional parameter
STANDALONE MSG AUTH	This parameter Enables/Disables the Standalone Messaging Service. If set to **0** the service is disabled. When set to **1** it is enabled. When set to **2** it is enabled for receiving Standalone Messages but not for sending.	Mandatory Parameter
IM CAP ALWAYS ON	If set to **1** (enabled), the Chat capability for all RCS contacts will be always reported as available and store and forward may be required for the message if the recipient cannot be reached. Otherwise (**0**), the capability will be reported based on the algorithm presented in section 2.7.1.1. When a service provider prefers that the client does not use Chat when an RCS contact is not currently registered in IMS, it sets the parameter to **0**.	Optional parameter (It is mandatory if CHAT AUTH is set to 1)

IM CAP NON RCS	This parameter configures the client to support chat with non-RCS contacts. If set to **1**, the Chat capability for all contacts will be always reported as available whether they are RCS enabled or not. Otherwise (**0**), the capability will be reported based on the setting for IM CAP ALWAYS ON and algorithm presented in section 2.7.1.1. For example, this can be used by Service Providers that are implementing the interworking of chat to SMS/MMS.	Optional parameter It is mandatory if CHAT AUTH is set and IM CAP ALWAYS ON is set to 1.
IM CAP NON RCS GROUP CHAT	This parameter defines whether and under which conditions the device is able to invite non RCS users in a Group Chat: (**0**): the device is not able to invite a non RCS user in any Group Chat session (default value) (**1**): the device is able to invite upon Group Chat creation or after Group Chat creation non RCS users in every Group Chat session (**2**): the device is able to invite upon Group Chat creation non RCS users only for the Group Chat sessions generated by its user	Optional parameter
IM CAP NON RCS LIMIT GROUP CHAT	This parameter controls the number of non RCS users that can be invited upon Group Chat creation by the user that generates the Group Chat. When set to 0 or not provided, there is no limit in the number of non RCS users that can be added.	Optional parameter
GROUP CHAT BREAKOUT ALLOWED PREFIXES	A list of prefixes of phone numbers used to identify the non RCS contacts that the client is allowed to add in a Group Chat. The prefix is interpreted by the client by matching the phone numbers of the address book or entered by the user starting from the left. The length of the prefix can be one or more digits and it can start with the "+" character. The service provider should take the subscriber's HPLMN numbering scheme into account when defining the prefixes. For the case that no prefixes are provided, all contacts can be invited to a Group Chat without restriction.	Optional parameter It is mandatory if IM CAP NON RCS GROUP CHAT is set to 1 or 2
IM SESSION AUTO ACCEPT	This parameter controls whether the client automatically accepts incoming session invitations (**1**) or whether acceptance depends on a user action (**0**) as defined through the IM SESSION START parameter. Automatic accept should only be used in a single device environment or if session forking on the AS is used.	Optional parameter (It is mandatory if CHAT AUTH is set to 1.)

IM SESSION START	This parameter defines the point in a chat when the receiver sends the 200 OK back to the sender confirming that the MSRP session can be established: **0** (RCS 6.x default): The 200 OK is sent when the receiver consumes the notification by opening the chat window. **1** (RCS Release 2-4 default): The 200 OK is sent when the receiver starts to type a message to be sent back in the chat window. **2**: The 200 OK is sent when the receiver presses the button to send a message (that is the message will be buffered in the client until the MSRP session is established). NOTE: as described in section 3.3.4, the parameter only affects the behaviour for a 1-to-1 session if no session between the parties has been established yet.	Optional parameter (It is mandatory if CHAT AUTH is set to 1.)
IM SESSION AUTO ACCEPT GROUP CHAT	This parameter controls whether the client automatically accepts incoming Group Chat session invitations (**1**) or whether acceptance depends on a user action (**0**) as defined through the IM SESSION START parameter. Automatic accept should only be used in a single device environment or if session forking on the AS is used.	Optional parameter (It is mandatory if CHAT AUTH is set to 1.)
IM SESSION TIMER	This parameter controls the time during which a 1-to-1 Chat session is allowed to be idle before it’s closed. When set to **0**, there shall be no timeout. The recommended value is 3 (three) minutes.	Optional parameter (It is mandatory if CHAT AUTH is set to 1)
MAX SIZE IM	This parameter controls the maximum size of the content sent within a chat session.	Optional parameter (It is mandatory if CHAT AUTH is set to 1.)
MAX SIZE STANDALONE	This parameter controls the maximum size of a message sent as a CPM Standalone message.	Optional parameter (It is mandatory if STANDALONE MSG AUTH is set to 1.)
MESSAGE STORE URL	The URL used to access the Message Store Server The parameter is optional and if not configured, means that the Service Provider is not deploying a Message Store server. Default value if not provided: msg-store.rcs. mnc<MNC>.mcc<MCC>.pub.3gppnetwork.org Whereby <MNC> and <MCC> shall be replaced by the respective values of the home network in decimal format and with a 2-digit MNC padded out to 3 digits by inserting a 0 at the beginning (as defined in [PRD-IR.67]).	Optional parameter

MESSAGE STORE USER / PASSWORD	The plain text user name and password for authentication with the Message Store Server. If the parameters are absent but the MESSAGE STORE AUTH parameter is present the following applies: • If the MESSAGE STORE AUTH parameter is set to "0" then the client shall use the values of "Realm User Name" and "Realm User Password" defined in Table 2 of [PRD-RCC.15] for authentication instead. • If the MESSAGE STORE AUTH parameter is set to "2" then the client shall derive user name and password from the GBA bootstrapped security association as defined in section 4.1.4.1.	Optional parameter.
MESSAGE STORE AUTH	This parameter controls the authentication mechanism used to access the Message Store Server. **0**: Plain User Name password via LOGIN command with user name and password from MESSAGE STORE USER / PASSWORD **2**: Authentication with B-TID and Ks_NAF derived from the GBA bootstrapped security association as defined in section 4.1.4.1. The client shall select the appropriate IMAP authentication mechanism from the server's IMAP CAPABILITY command response (i.e. LOGIN or AUTHENTICATE command). If not provided, this means that the Service Provider is not deploying a Message Store server.	Optional parameter
DATA CONNECTION SYNC TIMER	This parameter controls the time interval between two data connection state related triggered synchronizations. Unless timer is elapsed, the client shall not propagate any synchronisation request triggered due to change to the data connection state. When set to 0, timer is not activated and consequently all synchronization requests due to data connection state change shall be sent.	Optional parameter (It is mandatory if MESSAGE STORE AUTH is configured)
MESSAGE STORE SYNC TIMER	This parameter controls the time interval between two periodic client triggered synchronisations. Once any other type of synchronisation is triggered the timer is reset. When set to **0**, there shall be no automatic client request for synchronisation.	Optional parameter (It is mandatory if MESSAGE STORE AUTH is configured)

MESSAGE STORE EVENT REPORTING	This parameter is used to inform the Message Store Client whether to directly set flags in the Message Store or whether to indicate to the Messaging Server that it should set flags in the Message Store on behalf of the client. When set to **0** the client shall set flags in IMAP as needed via an IMAP connection (default). When set to **1**, Indicates that the client shall make use of the Event Reporting framework as described in section 4.1.4.8 when no IMAP connection exists so that the Messaging Server may set the flags in the Message Store on behalf of the client If not provided, the Message Store Client SHALL assume the same method as if value 0 had been specified.	Optional parameter
MESSAGE STORE ARCHIVE AUTH	This parameter Enables/Disables the Archive service. If set to **0** the Archive service is disabled. When set to **1** it is enabled and the client may archive messages. Default value is **0**.	Optional Parameter
SMS MESSAGE STORE	This parameter indicates to the client whether it shall store in the RCS Default folder any sent or received SMS. If this parameter is set to **0**, client shall not store any sent or received SMS/MMS. If this parameter is set to **1**, client shall store every sent and received SMS that cannot be correlated with the Common Message Store, If this parameter is set to **2**, client shall store every sent and received SMS and shall not attempt to correlate with the Common Message Store.	Optional parameter (It is mandatory if MESSAGE STORE AUTH is configured)
MMS MESSAGE STORE	This parameter indicates to the client whether it shall store in the RCS Default folder any sent or received MMS. If this parameter is set to **0**, client shall not store any sent or received MMS. If this parameter is set to **1**, client shall store every sent and received MMS that cannot be correlated with the Common Message Store, If this parameter is set to **2**, client shall store every sent and received MMS and shall not attempt to correlate with the Common Message Store.	Optional parameter (It is mandatory if MESSAGE STORE AUTH is configured)
CHAT MESSAGING TECHNOLOGY	This parameter allows selecting what technology is used for the chat service described in sections 3.3 and 3.4. If this parameter is set **0**, SIMPLE IM as specified in [RCS-SIMPLEIM-ENDORS] will be used. This is the default value if the parameter is not provided. If this parameter is set **1**, CPM as specified in [RCS-CPM-CONVFUNC-ENDORS].	Optional Parameter (It is mandatory if CHAT AUTH is set to 1.)
CHAT REVOKE TIMER	This parameter determines the maximum time between the client sending a Chat message and receiving its delivery notification. Once this timer expires without the client having received the delivery notification, the client shall automatically send a MessageRevoke request. For the case of a successful result, the user may be informed and the client shall fallback to SMS. When set to **0** (Default Value), sending MessageRevoke requests by the client is disabled.	Optional Parameter

Table 75: RCS additional Chat related configuration parameters

A.1.5. File Transfer related configuration

As there are no OMA defined parameters for File Transfer, this RCS specification includes only RCS specific parameters. These are described in the following table:

Configuration parameter	Description	RCS usage
PROVIDE FT	This parameter allows to enable (1) or disable (0) File Transfer via MSRP.	Mandatory Parameter
FT MAX SIZE	This is a file transfer size limit in Kilobyte (KB). If a file is bigger than FT MAX SIZE, the transfer will be cancelled automatically. This parameter is used to check for outgoing file size and is used to check for incoming file size if FT MAX SIZE INCOMING parameter is not provisioned. Please note that if it is set to **0**, this limit will not apply.	Optional parameter It is Mandatory if a PROVIDE FT is set to 1 or FT DEFAULT MECH is set to HTTP.
FT MAX SIZE INCOMING	This is a file transfer size limit in Kilobyte (KB) that a user can receive. If a receiving file is bigger than FT MAX SIZE INCOMING, the transfer will be cancelled automatically. Please note that if it is set to 0, this limit will not apply. This parameter may only be provisioned when the FT MAX SIZE parameter is not set to 0. When this parameter is not provisioned (i.e. no value or NULL), the device uses the size provisioned in FT MAX SIZE to check if an incoming file may be accepted. When this parameter is provisioned the device uses it to check if an incoming file may be accepted. NOTE: This optional parameter is used to override the max file size that can be received by the device to allow the flexibility of supporting different size of the files transfer by different interconnect operators. This parameter shall not be set to a smaller file size than FT MAX SIZE	Optional parameter
FT WARN SIZE	This is a file transfer size limit in KB to warn the user that a file transfer may end up in significant charges. Please note that if it is set to **0**, the user will not be warned.	Optional parameter It is Mandatory if PROVIDE FT is set to 1 or FT DEFAULT MECH is set to HTTP.

FT CAP ALWAYS ON	This parameter describes whether the file transfer via MSRP can take place independently of whether or not the receiving end is registered: (0 – *or not set*) File transfer depends on known capabilities of the recipient; (1) RCS Messaging Server based store and forward can be assumed to be enabled	Optional parameter; Mandatory if CHAT MESSAGING TECHNOLOGY is set to OMA CPM.
FT AUT ACCEPT	This parameter controls whether the client automatically accepts incoming File Transfer invitations (**1**) or whether acceptance depends on the user explicitly accepting (**0**). The parameter is only used if the file to be transferred is smaller than the limit configured in FT WARN SIZE. For files that are larger, the invitation will always require manual acceptance. Automatic accept should only be used in a single device environment or if session forking on the AS is used.	Optional parameter It is Mandatory if PROVIDE FT is set to 1 or FT DEFAULT MECH is set to HTTP.
FT HTTP CS URI	This parameter configures the URI of the HTTP content server where files will be uploaded by the originating side in case the destination cannot accept within the validity period. The parameter shall contain a full qualified URI. The URI should contain the "https" schema to enforce use of secure connections for the client's content server transactions. Default value if not provided: ftcontentserver.rcs.mnc<MNC>.mcc<MNC>.pub.3gppnetwork.org Whereby <MNC> and <MCC> shall be replaced by the respective values of the home network in decimal format and with a 2-digit MNC padded out to 3 digits by inserting a 0 at the beginning (as defined in [PRD-IR.67]).	Optional parameter
FT HTTP CS USER	This parameter is the name or identity that shall be used to authenticate the RCS client trying to either get a root URL (HTTP GET request) or upload a file (HTTP post request). If not provided and FT DEFAULT MECH is set to HTTP, the client shall use GBA authentication as described in section 3.5.4.8.3.	Optional parameter
FT HTTP CS PWD	This parameter is the password that shall be used to authenticate the RCS client trying to either get a root URL (HTTP GET request) or upload a file (HTTP post request). If not provided and FT DEFAULT MECH is set to HTTP, the client shall use GBA authentication as described in section 3.5.4.8.3.	Optional parameter
FT DEFAULT MECH	This parameter controls which file transfer mechanism (MSRP or HTTP) shall be used if both ends support both mechanisms Default value if not provided: MSRP	Optional parameter

FT HTTP FALLBACK	This parameter controls whether for a contact not supporting any of the enabled File Transfer Mechanisms multimedia content and files are to be sent using MMS (**0**, default value) or using the procedure defined in section 3.5.4.8.6 (**1**). NOTE: Even if this can be used without having FT DEFAULT MECH set to HTTP, FT HTTP CS URL, FT HTTP CS USER and FT HTTP CS PWD must be correctly set (or use the default values) when configuring the value 1 for this parameter.	Optional parameter

Table 76: RCS additional File Transfer related configuration parameters

A.1.6. Content Sharing related configuration

As there are no OMA defined parameters for content sharing, this RCS specification includes only RCS specific parameters. These are described in the following table:

Configuration parameter	Description	RCS usage
PROVIDE VS	This parameter allows to enable or to disable Video Share depending on network connectivity (only non-3GPP/non-3GPP2 networks, also on LTE, etc.).	Mandatory Parameter
PROVIDE IS	This parameter allows to enable (1) or disable (0) Image Share.	Mandatory Parameter
ALLOW VS SAVE	This parameter allows a Service Provider to configure whether a Video or Image Share session initiated by the RCS client can be saved or not. When set to (-1) the inclusion of the attribute defined in section 3.6.4.1.1 is up to user preference, when set to (0) the attribute will never be included, which is also the default handling if not provided, when set to (1) the attribute will always be included. NOTE: The parameter name includes VS for historic reasons.	Optional Parameter
VS MAX DURATION	This parameter enables the Service Provider of the inviting user's RCS client to control the maximum duration time of a Video Share session that the inviting user's RCS client is authorised to handle.	Optional parameter It is Mandatory if a PROVIDE VS is set to 1.
IS MAX SIZE	Maximum authorised size of the content that can be sent within an Image Share session. This parameter enables the Service Provider of the inviting user's RCS client to control the maximum size of the content that the inviting user's RCS client is authorised to send in an Image Share session.	Optional parameter It is Mandatory if a PROVIDE IS is set to 1.
COMPOSER AUTH	As per section 2.1.2 of [PRD-RCC.20]	As per section 2.1.2 of [PRD-RCC.20]
SHARED MAP AUTH	As per section 2.1.2 of [PRD-RCC.20]	As per section 2.1.2 of [PRD-RCC.20]

SHARED SKETCH AUTH	As per section 2.1.2 of [PRD-RCC.20]	As per section 2.1.2 of [PRD-RCC.20]
POST CALL AUTH	As per section 2.1.2 of [PRD-RCC.20]	As per section 2.1.2 of [PRD-RCC.20]
CALL COMPOSER TIMER IDLE	As per section 2.1.2 of [PRD-RCC.20]	As per section 2.1.2 of [PRD-RCC.20]

Table 77: RCS additional content sharing related configuration parameters

A.1.7. IMS Core / SIP related configuration

A.1.7.1. IMS Provisioning Parameters

IMS parameters shall be configured as defined in [PRD-RCC.15]. Not all parameters in the IMS Management Object defined in [3GPP TS 24.167] that is referred from [PRD-RCC.15] are relevant within the context of RCS though. Following table lists the parameters and their RCS usage:

Configuration parameter	Description	RCS usage
ConRef	Represents a network access point object	Mandatory parameter Not used for RCS Provided with dummy value if RCS is only user of MO: *dummy.apn* Selection of the APN for RCS depends on other parameters as defined in section 2.9.1.4
PDP_ContextOperPref	Indicates an operator’s preference to have a dedicated PDP context for SIP signalling.	Mandatory parameter Not used for RCS, always set to 0 (no preference) if RCS is only user of MO.

P-CSCF_Address	an FQDN or an IPv4 address to an IPv4 P-CSCF	Optional parameter Not provided if RCS is only user of MO. The P-CSCF address for RCS is provided in LBO_P-CSCF_Address
Timer_T1	Defines the SIP timer T1 – the RTT estimate	Mandatory parameter
Timer_T2	Defines the SIP timer T2 – the maximum retransmit interval for non-INVITE requests and INVITE responses.	Mandatory parameter
Timer_T4	Defines the SIP timer T4 – the maximum duration a message will remain in the network.	Mandatory parameter
Private_user_identity	Represents the private user identity	Mandatory parameter Used as defined in [PRD-RCC.15] and if applicable [PRD-NG.102]
Public_user_identity	Represents a public user identity.	Mandatory parameter Used as defined in [PRD-RCC.15] and if applicable [PRD-NG.102]
Home_network_domain_name	Indicates the operator’s home network domain.	Mandatory parameter Recommended to use ims.mnc<MNC>.mcc<MCC>.pub.3gppnetwork.org whereby <MNC> and <MCC> shall be replaced by the respective values of the home network in decimal format and with a 2-digit MNC padded out to 3 digits by inserting a 0 at the beginning (as defined in [PRD-IR.67]).

ICSI_List	Used to allow a reference to a list of IMS communication service identifiers that are supported by a subscriber's network for that subscriber.	Mandatory Tree Tree is not used for RCS, but mandatory to be provided. No leafs shall be provided if RCS is only user of MO.
LBO_P-CSCF_Address	Provides a reference to a list of P-CSCFs	Optional tree Mandatory for RCS Note: in a number of cases described in [PRD-NG.102] on a device enabled for VoLTE/VoWiFi the parameters provided in the tree should be ignored
Address (LBO_P-CSCF_Address)	Defines the FQDN, the IPv4 address or the IPv6 address of a P-CSCF	Mandatory parameter
AddressType (LBO_P-CSCF_Address)	Defines the type of address stored in the Address leaf node	Mandatory parameter
Resource_Allocation_Mode	Indicates whether UE initiates resource allocation for the media controlled by IM CN subsystem for all IMS sessions not covered by any "ICSI Resource Allocation Mode", when both UE and network can initiate resource allocation	Optional parameter Parameter not provided if RCS is only user of MO.
Voice_Domain_Preference_E_UTRAN	Indicates network operator's preference for selection of the domain to be used for voice communication services by the UE.	Optional parameter Parameter not provided if RCS is only user of MO.
SMS_Over_IP_Networks_Indication	Indicates network operator's preference for selection of the domain to be used for short message service (SMS) originated by the UE.	Optional parameter Parameter not provided if RCS is only user of MO.
Keep_Alive_Enabled	Indicates whether the UE sends keep alives	Mandatory parameter

Voice_Domain_Preference_UTRAN	Indicates network operator's preference for selection of the domain to be used for voice communication services by the UE.	Optional parameter Parameter not provided if RCS is only user of MO.
/<X>/Mobility_Management_IMS_Voice_Termination	Indicates whether the UE mobility management performs additional procedures to support terminating access domain selection by the network.	Optional parameter Parameter not provided if RCS is only user of MO.
RegRetryBaseTime	Represents the value of the base-time parameter	Optional parameter
RegRetryMaxTime	Represents the value of the max-time parameter	Optional parameter
PhoneContext_List	Used to allow a reference to a list of phone-context parameter values for other local numbers, than geo-local or home-local numbers	Optional tree Tree is not provided for RCS because only home-local and geo-local numbers are used (see section 2.5)
SS_domain_setting	Indicates the network operator's preference for the selection of the domain used by the UE when performing supplementary services (SS) setting control for voice services.	Optional parameter Parameter not provided if RCS is only user of MO.
PS_domain_IMS_SS_control_ preference	Provides a means to define the method for which Supplementary Services are controlled by the UE when SS setting control is to be invoked over the PS domain.	Optional parameter Parameter not provided if RCS is only user of MO.

Table 78: Usage of [3GPP TS 24.167] MO parameters for RCS

A.1.7.2. RCS Specific Provisioning parameters

This RCS specification includes the following additional IMS Core/SIP related configuration parameters:

Configuration parameter	Description	RCS usage
RCS VOLTE SINGLE REGISTRATION	This parameter defines the behaviour regarding the registration for RCS services, on whether it shall share the VoLTE registration or have a separate one. 0, the device *Shall* follow a dual registration approach (transition solution) 1 (default if not provided), the device *Shall* follow a single registration (target solution) 2, the device Shall follow a single registration for both RCS and VoLTE/ViLTE services when in the home network, and Shall follow a dual registration approach when roaming (transition solution). The parameter is also used to control the APN selection on devices that are not enabled for VoLTE as described in section 2.9.1.4.	Optional Parameter

Table 79: RCS additional IMS Core/SIP related configuration parameters

A.1.8. Geolocation related configuration

A.1.8.1. OMA SUPL Provisioning parameters

RCS does not rely on Secure User Plane Location (SUPL) for providing localisation social presence information, but extends the OMA Management Object for SUPL [SUPLMO] for the parameters relating to the location services.

Following table lists the OMA SUPL parameters and how they are handled for RCS.

Configuration parameter	Description	RCS usage
Addr	The address of the H-SLP	Optional parameter It is not provided for RCS
AddrType	The type of the address provided in Addr	Optional parameter It is not provided for RCS

Table 80: RCS usage of OMA SUPL configuration parameters

A.1.8.2. RCS Specific Provisioning parameters

This RCS specification includes the following additional geolocation related configuration parameters.

Configuration parameter	Description	RCS usage
PROVIDE GEOLOC PUSH	This parameter allows enabling (**1**) or disabling (**0**) the Geolocation PUSH service.	Mandatory Parameter
PROVIDE GEOLOC PULL	This parameter allows to disable (0) the Geolocation PULL service, or enable it with only File Transfer technology (2[38]).	Mandatory Parameter
GEOLOCATION TEXT MAX LENGTH	This parameter allows the control of the maximum length of the text describing the current location (with a maximum value of 200 characters).	Optional parameter It is mandatory and becomes relevant only if the Geolocation PULL or Geolocation PUSH service is available for the device.
GEOLOCATION VALIDITY	This parameter allows controlling the maximum time during which a location information should be considered valid.	Optional parameter. If present, it indicates A maximum value the user is authorised to enter.
GEOLOCATION PULL BLOCK TIMER	The interval during which the Geolocation PULL application is not allowed to send a PULL request to a target contact if a previous request was explicitly rejected	Optional Parameter It is mandatory and becomes relevant only if the Geolocation PULL with File Transfer service is available for the device.

Table 81: RCS additional geolocation related configuration parameters

A.1.9. Configuration related with Address book Back-up/Restore

This RCS specification does not include any additional address book back-up/restore related configuration parameters.

A.1.10. Capability discovery related configuration

This RCS specification includes the following RCS Specific configuration parameters related to the capability discovery:

[38] The value of 1 is not used for historic reasons

Configuration parameter	Description	RCS usage
DISABLE INITIAL ADDRESS BOOK SCAN	This parameter controls whether the device/client performs a capability check for all contacts in the address book when it is first started. When set to **0** (Default value), the device/client shall perform the scan When set to **1**, the device shall skip the scan and only perform capability exchange requests based on the other triggers defined in section 2.6. Note: this parameter will just disable the initial address book scan. It will not affect the Capability Polling which is controlled through the POLLING PERIOD parameter.	Optional parameter
POLLING PERIOD	This is the frequency in seconds at which to run a periodic capabilities update for all the contacts in the phone's address book whose capabilities are not available (such as non-RCS users) or are expired (see CAPABILITY INFO EXPIRY parameter). Please note that if set to **0**, this periodic update is not/no longer performed. However, the initial polling of the complete address book, as described in section 2.6.2 shall be done at the first configuration.	Optional parameter (It is mandatory when CAPABILITY DISCOVERY MECHANISM is enabled)
POLLING RATE	This parameter allows controlling the maximum rate (capability requests per POLLING RATE PERIOD) at which SIP OPTIONS and Presence Fetch operations are performed for the initial address book polling and potential subsequent pollings. It therefore provides some control over the network load caused when performing a capability discovery for the whole address book. When provided, the value has to be greater than 0. When not provided, the rate at which operations are performed for the initial address book polling is up to device implementation.	Optional parameter (It is mandatory if POLLING PERIOD is set to a value greater than 0.)
POLLING RATE PERIOD	This parameter allows defining the window in seconds over which the configured polling rate should be measured. When provided, the value has to be greater than 0.	Optional parameter (It is mandatory if POLLING RATE is set)
CAPABILITY INFO EXPIRY	When using the capability discovery mechanism and with the aim of minimising the traffic, an expiry time is set in the capability information fetched using SIP OPTIONS or Presence. When performing a whole address book capability discovery (i.e. polling), a capability query takes place only if the time since the last capability update took place is greater than this expiration parameter. Default value: 2592000 (30 days)	Optional parameter (It is mandatory if POLLING PERIOD is set to a value greater than 0.)

CAPABILITY DISCOVERY MECHANISM	This parameter allows selecting the default capability and new user discovery mechanism. If set to OPTIONS (**0**), the default mechanism employed for capability discovery and new users will be OPTIONS. If set to PRESENCE (**1**), the mechanism employed for capability discovery and new users will relay presence-based information. If not provided or set to OFF (**2**) the capability discovery mechanism is disabled.	Optional parameter
CAPABILITY DISCOVERY VIA COMMON STACK	This parameter allows selecting whether the device will fall back to OPTIONS if a discovery using presence fails with an error indicating that the other user does not support a presence based capability check. When set to **1**, this fallback is done. When set to **0** it is not done.	Optional parameter It is mandatory if CAPABILITY DISCOVERY MECHANISM is set to PRESENCE.
CAPABILITY DISCOVERY ALLOWED PREFIXES	A list of prefixes of phone numbers used to identify the contacts that are considered for the capability discovery mechanism. In case no prefix is included, capability discovery applies to all contacts. The prefix is interpreted by the client by matching the phone numbers of the address book or entered by the user starting from the left. The length can be one or more digits and it can start with the "+" character. The service provider should take the subscriber's HPLMN numbering scheme into account when defining the prefixes.	Optional parameter
NON RCS CAPABILITY INFO EXPIRY	This parameter allows to better control the amount of capability query sent to non RCS contacts. When updating a capability for a non RCS contact, a capability query takes place only if the time since the last capability update took place is greater than this parameter. Default value: 2592000 (30 days)	Optional parameter

Table 82: RCS additional capability discovery related configuration parameters

A.1.11. APN configuration

This RCS specification includes the following RCS Specific configuration parameters targeting APN configuration (see sections 2.9.1.4 and 2.13):

Configuration parameter	Description	RCS usage
NO MSRP SUPPORT	This parameter is used to list roaming partners the home operator knows do not support MSRP. When MSRP is not supported when roaming, the client shall use the HOS APN for RCS if obtained through client configuration; otherwise, shall use the Internet APN. If not instantiated, the device shall behave as if it was an empty list. NOTE: a device that cannot support the IMS APN shall ignore this parameter	Optional Parameter

Table 83: RCS roaming configuration parameters

A.1.12. IP Voice and Video Call configuration

As there are no OMA defined parameters for IP Voice and Video Call, this RCS specification includes only RCS specific parameters. These are described in the following table:

Configuration parameter	Description	RCS usage
PROVIDE IR94 VIDEO	This parameter allows to enable (1) or disable (0) IR94 Video Calling	Optional Parameter. This parameter SHOULD be set if IR94 client is present on UE
PROVIDE RCS IP VOICE CALL	This parameter allows to enable or to disable the RCS IP Voice Call Service on non-VoLTE/VoWiFi enabled primary devices for use over non-cellular access and on secondary devices regardless of the access.	Mandatory Parameter
PROVIDE RCS IP VIDEO CALL	This parameter allows to enable or to disable the RCS IP Video Call Service on devices depending on network connectivity (only non-3GPP/non-3GPP2 networks, also on LTE, etc.).	Mandatory Parameter
PROVIDE IR51 VOICE	This parameter allows to enable (**1**) or disable (**0**) IR.51 Voice Calling	Mandatory parameter
PROVIDE IR51 VIDEO	This parameter allows to enable (**1**) or disable (**0**) IR.51 Video Calling. This parameter can only be set to enabled (**1**) if PROVIDE IR51 VOICE is also set to enabled (**1**).	Mandatory parameter

Table 84: RCS IP Voice and Video Call configuration parameters

A.1.13. Service Provider specific extensions

A Service Provider may provide Service Provider specific extensions to the configuration parameters. This can be done both at the individual service level and add the global level (e.g. for the configuration of Service Provider specific services). All parameters are optional and if provided may be ignored by clients that are not Service Provider specific.

A.1.14. Extensions configuration parameters

The RCS specification includes the following additional Extensions related configuration parameters:

Configuration parameter	Description	RCS usage
ALLOW RCS EXTENSIONS	This parameter indicates to a device whether Extensions using the RCS infrastructure are allowed or not. If this parameter is set to: 0, Not Allowed: The SIP REGISTER shall NOT include the IARIs pertaining to Extensions, and shall NOT include the ICSI for RCS Extension to Extension in its Contact header. Also, they are not included as part of capability discovery. 1, Allowed: The SIP REGISTER shall include IARIs pertaining to Extensions, and the ICSI for RCS Extension to Extension if used by installed applications, in its Contact header. Also, they are included as part of capability discovery.	Mandatory Parameter
EXTENSIONS MAX MSRP SIZE	This parameter controls the maximum size of the MSRP content sent within a 1-to-1 RCS Extension to Extension session. A value of **0** (zero) means the maximum size of content sent is unlimited.	Optional parameter (It is mandatory and becomes relevant only if ALLOW RCS EXTENSIONS is set to **1**

Table 85: RCS extensions configuration parameters

NOTE: Parameters controlling the use of the terminal APIs on an RCS device may be defined in [PRD-RCC.53]

A.1.15. Data Off

The RCS specification includes following configuration parameters controlling the behaviour of the respective services when connected over cellular networks using a primary device that is not using the internet APN for RCS (see section 2.9.1.4) and data is switched off by the user:

Configuration parameter	Description	RCS usage
RCS MESSAGING DATA OFF	This parameter indicates whether the 1-to-1 and Group Chat, Standalone Messaging and Geolocation PUSH services *Should* remain available in case the cellular data switch is switched off (either toggled manually by the user or automatically during roaming). When set to 0 the Chat, Standalone Messaging and Geolocation PUSH services are disabled on cellular networks when cellular data is switched off. When set to 1 the Chat, Standalone Messaging and Geolocation PUSH services remain available on cellular networks when cellular Data is switched off. When set to 2 the Chat, Standalone Messaging and Geolocation PUSH services remain available on cellular networks when cellular Data is switched off and the device is attached to the HPLMN, otherwise Chat, Standalone Messaging and Geolocation PUSH services are disabled.	Optional Parameter It becomes mandatory if CHAT AUTH and/or STANDALONE MSG AUTH is set to 1 or 2 (see A.1.4.3)
FILE TRANSFER DATA OFF	This parameter indicates whether the File Transfer service *Should* remain available in case the cellular data switch is switched off (either toggled manually by the user or automatically during roaming). When set to 0 the File Transfer Service is disabled on cellular networks when cellular data is switched off. When set to 1 the File Transfer service remains available on cellular networks when cellular Data is switched off. When set to 2 the File Transfer service remains available on cellular networks when cellular Data is switched off and the device is attached to the HPLMN, otherwise the File Transfer Service is disabled. If File Transfer is available when cellular Data is switched off, the File Transfer technology selection defined in section 3.5 applies. For File Transfer via MSRP the parameter controls the complete File Transfer traffic. For File Transfer via HTTP the parameter controls the HTTP transactions for upload and download of files to/from the HTTP Content Server. The transfer of the file-info for File Transfer via HTTP via RCS messaging is controlled through the RCS MESSAGING DATA OFF parameter.	Optional Parameter It becomes mandatory if PROVIDE FT is set to 1 or FT DEFAULT MECH is set to HTTP (see A.1.5)

Configuration parameter	Description	RCS usage
SMSOIP DATA OFF	This parameter indicates whether the SMS over IP service *Should* remain available in case the cellular data switch is switched off (either toggled manually by the user or automatically during roaming). When set to **0** the SMS over IP service is disabled when cellular data is switched off. When set to **1** the SMS over IP service remains available when cellular data is switched off. When set to **2** the SMS over IP service remains available when cellular data is switched off and the device is attached to the HPLMN, otherwise the SMS over IP service is disabled. The client shall enable SMS over CS, SMS over GPRS or SMS over SGs as applicable.	Optional Parameter It becomes mandatory if SMS_Over_IP_Networks_Indication is set to 1 (see [3GPP TS 24.167]).
MMS DATA OFF	This parameter indicates whether MMS *Should* remain available in case the cellular data switch is switched off (either toggled manually by the user or automatically during roaming). When set to **0** MMS is disabled when cellular data is switched off. When set to **1** (default value) MMS remains available when cellular Data is switched off. When set to **2** MMS is remains available when cellular data is switched off and the device is attached to the HPLMN, otherwise MMS is disabled. NOTE: the device’s settings to enable/disable automatic download of received MMS messages remain applicable.	Optional Parameter
CONTENT SHARE DATA OFF	This parameter indicates whether the Video Share, Image Share, Shared Map and Shared Sketch services *Should* remain available in case cellular data is switched off (either toggled manually by the user or automatically during roaming). When set to **0** the Video Share, Image Share, Shared Map and Shared Sketch services are disabled on cellular networks when cellular data is switched off. When set to **1** the Video Share, Image Share, Shared Map and Shared Sketch services remain available on cellular networks when cellular data is switched off. When set to **2** the Video Share, Image Share, Shared Map and Shared Sketch services remain available on cellular networks when cellular data is switched off and the device is attached to the HPLMN, otherwise the Video Share, Image Share, Shared Map and Shared Sketch services are disabled.	Optional Parameter It becomes mandatory if PROVIDE VS and/or PROVIDE IS and/or SHARED MAP AUTH (see section 2.1.2 of [PRD-RCC.20]) and/or SHARED SKETCH AUTH (see section 2.1.2 of [PRD-RCC.20]) is set to 1 is set to 1 (see A.1.6)

Configuration parameter	Description	RCS usage
PRE AND POST CALL DATA OFF	This parameter indicates whether the Call Composer and Post Call services Should remain available in case cellular data is switched off (either toggled manually by the user or automatically during roaming). When set to **0** the Call Composer and Post-call services are disabled on cellular networks when cellular data is switched off. When set to **1** the Call Composer and Post-call services remain available on cellular networks when cellular data is switched off. When set to **2** the Call Composer and Post-call services remain available on cellular networks when cellular data is switched off and the device is attached to the HPLMN, otherwise the Call Composer and Post-call services are disabled.	Optional Parameter It becomes mandatory if CALL COMPOSER AUTH and/or POST CALL AUTH is set to 1 (see section 2.1.2 of [PRD-RCC.20])
VOLTE DATA OFF[39]	This parameter indicates whether the VoLTE service (as defined in [PRD IR.92] and [PRD IR.58]) *Should* remain available in case the cellular data switch is switched off (either toggled manually by the user or automatically during roaming). When set to **0** the VoLTE service is disabled when cellular data is switched off. When set to **1** the VoLTE service remains available when cellular data is switched off. When set to **2** the VoLTE service is disabled when cellular data is switched off and the device is attached to the HPLMN, otherwise the VoLTE service is disabled. If the VoLTE service is disabled the supplementary service management via the Ut interface shall not be available. Note: This configuration parameter does not impact the operation of Multimedia Telephony if applied over EPC integrated Wi-Fi.	Optional Parameter It becomes mandatory on devices supporting VoLTE as defined in [PRD-IR.92] if Voice_Domain_Preference_E_UTRAN is set to 1 (see [3GPP TS 24.167]).

[39] In the future IR.92 or other PRDs may include parameters providing similar behaviour. In that case in case of conflict that parameter which would be applicable beyond the scope of devices implementing RCS shall get priority over this one when conflicting values are configured.

Configuration parameter	Description	RCS usage
IP VIDEO CALL DATA OFF[40]	This parameter indicates whether the IP Video Call service (see section 3.9, either provided as Video Call over LTE service defined in [PRD IR.94] or as RCS IP Video Call defined in section 3.9.4.1) *Should* remain available in case the cellular data switch is switched off (either toggled manually by the user or automatically during roaming). When set to **0** the IP Video Call service is disabled when cellular data is switched off. When set to **1** the IP Video Call service remains available even when cellular data is switched off. When set to **2** the IP Video Call service remains available when cellular data is switched off and the device is attached to the HPLMN, otherwise the IP Video Call service is disabled. If IP Video Call is disabled the device *Shall Not* include the *video* media tag in the SIP INVITE requests and responses exchanged for VoLTE calls when the cellular data switch is switched off.	Optional Parameter It becomes mandatory on devices supporting VOLTE if VOLTE DATA OFF is set to 1 and PROVIDE IR94 is set to 1 (see A.1.12) or on devices where PROVIDE RCS IP VIDEO CALL is set to a value that is greater than 1 (see A.1.12).
EXTENSIONS DATA OFF	This parameter indicates whether the Extensions to Extension service (see section 3.12.2) *Should* remain available in case the cellular data switch is switched off (either toggled manually by the user or automatically during roaming). When set to **0** (Default value) the Extensions to Extension service is disabled on cellular networks when cellular data is switched off. When set to **1** the Extensions to Extension service remains available on cellular networks when cellular data is switched off. When set to **2** the Extensions to Extension service is enabled on cellular networks when cellular data is switched off and the device is attached to the HPLMN, otherwise Extensions to Extension service is disabled.	Optional Parameter

[40] In future IR.94 or other PRDs may include parameters providing similar behaviour. In that case in case of conflict that parameter which would be applicable beyond the scope of devices implementing RCS shall get priority over this one when conflicting values are configured.

Configuration parameter	Description	RCS usage
PROVISIONING DATA OFF	This parameter indicates whether the Device Provisioning (see section 2.3) *Should* remain available in case the cellular data switch is switched off (either toggled manually by the user or automatically during roaming). When set to **0** device provisioning is disabled on cellular networks when cellular data is switched off. When set to **1** (default value) device provisioning remains available on cellular networks even when cellular data is switched off. When set to **2** device provisioning is disabled on cellular networks other than the HPLMN when cellular data is switched off, otherwise Device Provisioning is enabled NOTE: when device provisioning is disabled in cellular data access for the client it is only possible to change the value when the device provisioning becomes enabled again, e.g. by connecting to Wi-Fi.	Optional parameter
SYNC DATA OFF	This parameter indicates whether the synchronization with the Common Message Store Should remain available in case the cellular data is switched off (either toggled manually by the user or automatically during roaming). When set to **0** the synchronisation with the Common Message Store is disabled when cellular data is switched off. When set to **1** synchronisation with the Common Message Store remains available when cellular data is switched off. When set to **2** the synchronisation with the Common Message Store remains available when cellular data is switched off and the device is attached to the HPLMN, otherwise the synchronisation with the Common Message Store is disabled. If the synchronisation with the Common Message Store is disabled, the client shall not invoke the synchronisation with the Common Message Store, shall not set message flags via IMAP and event notification framework and shall not store SMS and MMS messages.	Optional Parameter It becomes mandatory if MESSAGE STORE AUTH is present (see A.1.4.3)

Table 86: RCS Data Off Configuration Parameters

NOTE1: No parameter is provided for RCS IP Voice Calls because for primary devices they are only available on the Wi-Fi bearer.

NOTE2: These parameters only affect behaviour on cellular networks. Services that can be offered over non-cellular networks remain available over such networks irrespective of the setting of the cellular data switch. These parameters also affect services when RCS uses the HOS APN.

A.1.16. RCS VV-Mail configuration parameters

This RCS specification includes the following configuration parameters related to the RCS VV-Mail:

Configuration parameter	Description	RCS usage
RCS VV-MAIL AUTH	This parameter Enables/Disables the RCS VV-MAIL service. If set to **0** the RCS VV-Mail service is disabled. When set to **1** it is enabled. If not present, the RCS VV-MAIL service is disabled by default.	Optional Parameter
EXTENSIONS MAX MSRP SIZE	This parameter Enables/Disables the RCS VV-MAIL and VMS sync. If set to **0** the RCS VV-Mail and VMS sync is disabled. When set to **1** it is enabled. If not present, the RCS VV-MAIL and VMS sync is disabled by default. NOTE: RCS VV-MAIL VMS SYNC is only applicable to secondary devices. If present, it should be ignored by the primary device.	Optional parameter

Table 87: RCS VV-Mail configuration parameters

A.2. RCS Management trees additions

Please note that all the configuration subtrees described in this section have as type property for the root nodes (that is the /<X> root nodes) urn:gsma:mo:rcs:6.0. All RCS specific MOs shall be placed in this RCS subtree:

Node: /<x>

Under this interior node the RCS parameters that belong to RCS specific MOs are placed

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 88: RCS MO sub tree addition node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs:6.0*
- Associated HTTP XML characteristic type: N/A

The DM Client assigns a unique name for the <x> node which consists of *[Parental Node]-Index* (e.g. *Ext/<x>/ChatAuth -> Ext/Ext-1/ChatAuth*). The index 1 belongs to the actual SIM card.

The following alert type MUST be used in a Generic Alert [DMPRO] message sent by the DM client in case of a client initiated management session towards the DM server related to an RCS MO:

- urn:gsma:mo:rcs:6.0:provision

The alert type is used to identify the operation that needs to be performed on the device and identifies the current version of the RCS Management Object.

A.2.1. Services sub tree additions

This RCS specification includes the following additions as a new services sub tree, the Services MO sub tree. Please note this sub tree is not included in any other specifications. So no other nodes from those specifications need to be added:

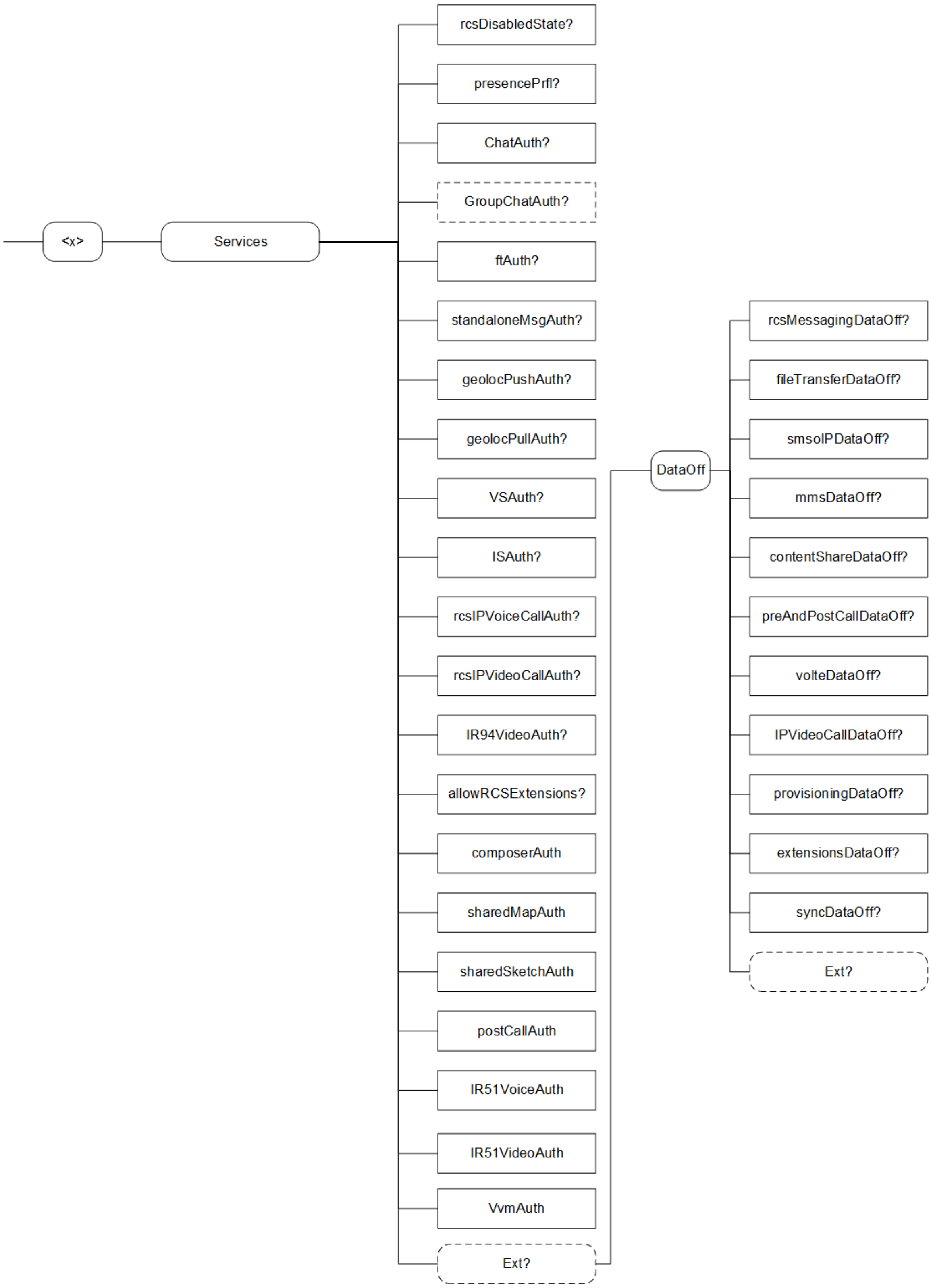

Figure 120: RCS additions, Services sub tree

The associated HTTP configuration XML structure is presented in the table below:

```
<characteristic type="SERVICES">
	<parm name="presencePrfl" value="X"/>
	<parm name="ChatAuth" value="X"/>
	<parm name="GroupChatAuth" value="X"/>
	<parm name="ftAuth" value="X"/>
	<parm name="standaloneMsgAuth" value="X"/>
	<parm name="geolocPullAuth" value="X"/>
	<parm name="geolocPushAuth" value="X"/>
	<parm name="vsAuth" value="X"/>
	<parm name="isAuth" value="X"/>
	<parm name="rcsIPVoiceCallAuth" value="X"/>
	<parm name="rcsIPVideoCallAuth" value="X"/>
	<parm name="IR94VideoAuth" value="X"/>
	<parm name="allowRCSExtensions" value="X"/>
	<parm name="composerAuth" value="X"/>
	<parm name="sharedMapAuth" value="X"/>
	<parm name="sharedSketchAuth" value="X"/>
	<parm name="postCallAuth" value="X"/>
	<parm name="IR51VoiceAuth" value="X"/>
	<parm name="IR51VideoAuth" value="X"/>
	<parm name="VvmAuth" value="X"/>
	<characteristic type="Ext">
		<characteristic type="DataOff">
			<parm name="rcsMessagingDataOff" value="X"/>
			<parm name="fileTransferDataOff" value="X"/>
			<parm name="smsoIPDataOff" value="X"/>
			<parm name="mmsDataOff" value="X"/>
			<parm name="contentShareDataOff" value="X"/>
			<parm name="preAndPostCallDataOff" value="X"/>
			<parm name="volteDataOff" value="X"/>
			<parm name="IPVideoCallDataOff" value="X"/>
			<parm name="provisioningDataOff" value="X"/>
			<parm name="extensionsDataOff" value="X"/>
			<parm name="syncDataOff" value="X"/>
			<characteristic type="Ext"/>
		</characteristic>
	</characteristic>
</characteristic>
```

Table 89 : Services MO sub tree associated HTTP configuration XML structure

Note: rcsDisabledState is not included in Table 89 since the presence of that parameter would invalidate most of the other parameters included.

Node: /<x>/Services

Under this interior node the RCS parameters related to the enabling/disabling of services are placed

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 90: Services MO sub tree addition services node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-services:6.0*

- Associated HTTP XML characteristic type: “SERVICES”

Node: /<x>/Services/rcsDisabledState

Leaf node that controls the state of the RCS services on the device

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

Table 91: Services MO sub tree addition parameters (rcsDisabledState)

- Values:
  0, the client is temporarily disabled
  -1, the client is permanently disabled
  -2, the client is permanently disabled, but user triggered events result in an attempt to re-enable the client
  -3, the client is placed in dormant state
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter: “rcsDisabledState”

NOTE: Given that due to the definition in section A.1.1, this parameter and the other parameters in this management object are mutually exclusive it is not included in Table 89 and Table 220.

Node: /<x>/Services/presencePrfl

Leaf node that describes whether or not the social presence functionality is supported

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 92: Services MO sub tree addition parameters (presencePrfl)

- Values: If set to 1, it is supported. If set to 0, it is not supported.
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter: “presencePrfl”

Node: /<x>/Services/ChatAuth

Leaf node that represents the authorisation for the user to use the chat service

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 93: Services MO sub tree addition parameters (ChatAuth)

- Values: 0, 1
  0- Indicates that chat service is disabled
  1- Indicates that chat service is enabled
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “ChatAuth”

Node: /<x>/Services/GroupChatAuth

Leaf node that represents the authorisation for the user to use the group chat service

If not instantiated, the ChatAuth parameter shall control the authorisation for both 1-to-1 and Group Chat.

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

Table 94: Services MO sub tree addition parameters (GroupChatAuth)

- Values: 0, 1
  0- Indicates that Group Chat service is disabled
  1- Indicates that Group Chat service is enabled
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “GroupChatAuth”

Node: /<x>/Services/ftAuth

Leaf node that represent the authorisation for user to use the File Transfer service via MSRP

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 95: Services MO sub tree addition parameters (ftAuth)

- Values: 0, 1
  0- Indicates that File Transfer service via MSRP is disabled
  1- Indicates that File Transfer service via MSRP is enabled
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “ftAuth”

Node: /<x>/Services/standaloneMsgAuth

Leaf node that represents the authorisation for user to use the standalone messaging service

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 96: Services MO sub tree addition parameters (standaloneMsgAuth)

- Values: 0, 1, 2
  0- The standalone messaging service is not provided. SMS and MMS is used instead
  1- Sending and receiving of messages via the standalone messaging service is provided and uses CPM as specified in [RCS-CPM-CONVFUNC-ENDORS].
  2- Receiving of messages via the standalone messaging service is provided and uses CPM as specified in [RCS-CPM-CONVFUNC-ENDORS]. SMS and MMS are still used for sending.
- Post-reconfiguration actions:
  - If the value of the configuration parameter transits from value 0 to value 1 or 2, the client may wait till the next scheduled refresh re-REGISTER request or may issue re-REGISTER request immediately.
  - If the value of the configuration parameter transits from value 1 or 2 to value 0, the client may wait till the next scheduled refresh re-REGISTER request or may issue re-REGISTER request immediately.
  - If the value of the configuration parameter transits from value 1 to value 2, the client shall stop sending messages via the standalone messaging service.
  - If the value of the configuration parameter transits from value 2 to value 1, the client shall be able to send messages via the standalone messaging service.
- Associated HTTP XML parameter ID: "standaloneMsgAuth"

Node: /<x>/Services/geolocPullAuth

Leaf node that represents the authorisation for the user to use the Geolocation PULL service

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 97: Services MO sub tree addition parameters (geolocPullAuth)

- Values: 0, 2
  0- Indicates that Geolocation PULL service is disabled
  2[38]- Indicates that Geolocation PULL service is enabled only with File Transfer technology
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: "geolocPullAuth"

Node: /<x>/Services/geolocPushAuth

Leaf node that represents the authorisation for the user to use the Geolocation PUSH service

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 98: Services MO sub tree addition parameters (geolocPushAuth)

- Values: 0, 1
  0- Indicates that Geolocation PUSH service is disabled
  1- Indicates that Geolocation PUSH service is enabled
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “geolocPushAuth”

Node: /<x>/Services/VSAuth

Leaf node that represents the authorisation for user to use Video Share service

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 99: Services MO sub tree addition parameters (VSAuth)

- Values: an unsigned 32 bit integer value that is mapped to a bit array indicating the radio technologies in which Video Share can be initiated. The mapping is as follows from MSB to LSB:

31 MSB	…	5	4	3	2	1	0 LSB
Reserved	Reserved	Reserved	LTE	HSPA	3G	Wi-Fi	All coverages

Table 100: VSAuth value to radio technology mapping

Reserved bits should be ignored by the client.
Some examples of this mapping of values to radio technologies in which Video Share is supported (only least significant byte mentioned):
xxx00000b (i.e. 0)- Indicates that the Video Share service is disabled
xxx00001b (i.e. 1) - Indicates that the Video Share service is enabled for all coverages
xxx00010b (i.e. 2)- Indicates that the Video Share service is enabled for non-3GPP/non-3GPP2 access only (e.g. Wi-Fi, xDSL)
xxx10000b (i.e. 16)- Indicates that the Video Share service is enabled for LTE access only
xxx10010b (i.e. 18)- Indicates that the Video Share service is enabled for non-3GPP/non-3GPP2 access (e.g. Wi-Fi, xDSL) and for LTE access
xxx11000b (i.e. 24)- Indicates that the Video Share service is enabled for LTE/HSPA access only

xxx11010b (i.e. 26)- Indicates that the Video Share service is enabled for non-3GPP/non-3GPP2 access (e.g. Wi-Fi, xDSL) and for LTE/HSPA access
xxx11100b (i.e. 28)- Indicates that the Video Share service is enabled for 3G, HSPA and LTE
xxx11110b (i.e. 30)- Indicates that the Video Share service is enabled for Wi-Fi, 3G, HSPA and LTE cellular access

- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “vsAuth”

Node: /<x>/Services/ISAuth

Leaf node that represents the authorisation for user to use Image Share service

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 101: Services MO sub tree addition parameters (ISAuth)

- Values: 0, 1
  0- Indicates that Image Share service is disabled
  1- Indicates that Image Share service is enabled
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “isAuth”

Node: /<x>/Services/rcsIPVoiceCallAuth

Leaf node that represents the authorisation for user to use RCS IP Voice Call service when the a non-VoLTE/VoWiFi enabled primary device is using non-cellular access or a secondary device is using any access

NOTE: when the device is in another mode the RCS IP Voice Call Service should not be provided regardless of the access network used for RCS

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 102: Services MO sub tree addition parameters (rcsIPVoiceCallAuth)

- Values: 0 or 1:
- 0- Indicates that the RCS IP Voice Call service is disabled on devices when in the applicable coverage
  1- Indicates that the RCS IP Voice Call service is enabled on devices when in the applicable coverage

- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “rcsIPVoiceCallAuth”

Node: /<x>/Services/rcsIPVideoCallAuth

Leaf node that represents the authorisation for user to use the RCS IP Video Call service

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 103: Services MO sub tree addition parameters (rcsIPVideoCallAuth)

- Values: an unsigned 32 bit integer value that is mapped to a bit array indicating the radio technologies in which an RCS IP Video Call can be initiated. The mapping is as follows from MSB to LSB:

31 MSB	…	4	3	2	1	0 LSB
Reserved	Reserved	Reserved	LTE	HSPA	3G	Wi-Fi

Table 104: rcsIPVideoCallAuth value to radio technology mapping

Reserved bits should be ignored by the client.

NOTE: For established calls, the call should be continued as long as there is IP continuity and the available bandwidth allows.

Some examples of this mapping of values to radio technologies in which RCS IP Video Calls are supported (only least significant byte mentioned):
xxxx0000b (i.e. 0)- Indicates that the RCS IP Video Call service is disabled
xxxx0001b (i.e. 1)- Indicates that the RCS IP Video Call service is enabled for non-3GPP/non-3GPP2 access only (e.g. Wi-Fi, xDSL)
xxxx1000b (i.e. 8)- Indicates that the IP Video Call service is enabled for LTE access only
xxxx1001b (i.e. 9)- Indicates that the RCS IP Video Call service is enabled for non-3GPP/non-3GPP2 access (e.g. Wi-Fi, xDSL) and for LTE access
xxxx1100b (i.e. 12)- Indicates that the RCS IP Video Call service is enabled for LTE/HSPA access only
xxxx1101b (i.e. 13)- Indicates that the RCS IP Video Call service is enabled for non-3GPP/non-3GPP2 access (e.g. Wi-Fi, xDSL) and for LTE/HSPA access
xxxx1110b (i.e. 14)- Indicates that the RCS IP Video Call service is enabled for 3G, HSPA and LTE
00001111b (i.e. 15)- Indicates that the RCS IP Video Call service is enabled for Wi-Fi, 3G, HSPA and LTE cellular access

- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.

- Associated HTTP XML parameter ID: “rcsIPVideoCallAuth”

Node: /<x>/Services/IR94VideoAuth

Leaf node that represents the authorisation for user to use IR94 Video Calling service

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 105: Services MO sub tree addition parameters (IR94VideoAuth)

- Values: 0, 1
  0- Indicates that IR94 Video Calling service is disabled
  1- Indicates that IR94 Video Calling service is enabled
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “IR94VideoAuth”

Node: /<x>/Services/allowRCSExtensions

Leaf node that describes whether use of RCS Extensions by the RCS Client is allowed.

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 106: Services MO sub tree addition parameters (allowRCSExtensions)

- Values: 0, 1
  0- Not Allowed: The SIP REGISTER shall NOT include the IARIs pertaining to Extensions, and shall NOT include the ICSI for RCS Extension to Extension in its Contact header (default value).
  1- Allowed: The SIP REGISTER shall include IARIs pertaining to Extensions, and the ICSI for RCS Extension to Extension if used by installed applications, in its Contact header.
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “allowRCSExtensions”

Node: /<x>/Services/composerAuth

As per 2.1.2.1 of [PRD-RCC.20]

Node: /<x>/Services/sharedMapAuth

As per 2.1.2.1 of [PRD-RCC.20]

Node: /<x>/Services/sharedSketchAuth

As per 2.1.2.1 of [PRD-RCC.20]

Node: /<x>/Services/postCallAuth

As per 2.1.2.1 of [PRD-RCC.20]

Node: /<x>/Services/IR51VoiceAuth

Leaf node that represents the authorisation for user to use IR51 Voice Calling service

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 107: Services MO sub tree addition parameters (IR51VoiceAuth)

- Values: 0, 1
  0, Indicates that IR51 Voice Calling service is disabled
  1, Indicates that IR51 Voice Calling service is enabled
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “IR51VoiceAuth”

Node: /<x>/Services/IR51VideoAuth

Leaf node that represents the authorisation for user to use IR51 Voice Calling service

The node shall be instantiated if the rcsDisabledState node is not provided.

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

Table 108: Services MO sub tree addition parameters (IR51VideoAuth)

- Values: 0, 1
  0, Indicates that IR51 Video Calling service is disabled
  1, Indicates that IR51 Video Calling service is enabled
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “IR51VideoAuth”

Node: /<x>/Services/VvmAuth

Leaf node that represent the authorisation of the RCS VV-Mail service.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	bool	Get

Table 109: Services MO sub tree addition parameters (VvmAuth)

- Values: 0, 1
  0- The value indicates that the voice mail service is disabled (default)
  1- The value indicates that the voice mail service is enabled
- Post-reconfiguration actions: As the client remains unregistered during configuration, there are no additional actions apart from re-registering using the new parameter if the parameter value has changed.
- Associated HTTP XML parameter ID: “VvmAuth”

Node: /<x>/Services/Ext

An extension node for Service Provider specific parameters. Clients that are not aware of any extensions in this subtree (e.g. because they are not Service Provider specific) should not instantiate this tree.

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	Node	Get

Table 110: Services MO sub tree addition Service Provider Extension Node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-services:6.0:Ext*
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML characteristic type: “EXT”

Node: /<x>/Services/Ext/DataOff

Under this interior node where the specific RCS parameters are placed that relate to the services behaviour on cellular networks when the cellular data switch is switched off.

It shall be instantiated for primary devices where it is required to be supported.

NOTE: This tree is included as part of the ext tree rather than sitting directly under Services for historic reasons

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	Node	Get

Table 111: Data Off Services Extension MO sub tree addition node

- Values: N/A
- Type property of the node is: urn:gsma:mo:rcs-services:6.0:Ext:DataOff
- Associated HTTP XML characteristic type: “DataOff”

Node: /<x>/Services/Ext/DataOff/rcsMessagingDataOff

Controls the Chat, Standalone Messaging and Geolocation PUSH service behaviour when the cellular data switch is switched off.

The parameter is only applicable in case the Chat or Standalone services are supported. It will not be instantiated otherwise.

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

Table 112: Data Off Services Extension MO sub tree addition parameters (rcsMessagingDataOff)

- Values:
  0: the Chat, Standalone Messaging and Geolocation PUSH services are not available on cellular networks when cellular data is switched off
  1: the Chat, Standalone Messaging and Geolocation PUSH services are available on cellular networks when cellular data is switched off
  2: the Chat, Standalone Messaging and Geolocation PUSH services are available on cellular networks when cellular data is switched off and the device is attached to the HPLMN, otherwise the services are disabled.
- Post-reconfiguration actions: If the value of the configuration parameter transits from 0 to 1 or from 0 to 2 while the device is connected to the HPLMN and at least one RCS messaging service is authorised the client shall (re-)register in IMS to add the relevant media feature tags for Chat, File Transfer, Standalone Messaging and Geolocation PUSH services according to the authorisation of these services.
  If the value of the configuration parameter transits from 1 to 0 or from 2 to 0 while the device is connected to a cellular access network other than the HPLMN and at least one of the RCS messaging services is registered in IMS, the client shall de- or re-register with in IMS to remove the media feature tags for Chat, File Transfer, Standalone Messaging or Geolocation PUSH services if these have been registered.
- Associated HTTP XML characteristic type: “rcsMessagingDataOff”

Node: /<x>/Services/Ext/DataOff/fileTransferDataOff

Controls the File Transfer service behaviour when the cellular data switch is switched off.

The parameter is only applicable in case the File Transfer service is supported. It will not be instantiated otherwise.

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

Table 113: Data Off Services Extension MO sub tree addition parameters (fileTransferDataOff)

- Values:
  0: the File Transfer service is not available on cellular networks when cellular data is switched off
  1: the File Transfer service is available on cellular networks when cellular data is switched off
  2: the File Transfer service is available on cellular networks when cellular data is switched off and the device is attached to the HPLMN, otherwise the service is disabled.
- Post-reconfiguration actions: If the value of the configuration parameter transits from 0 to 1 or from 0 to 2 while the device is connected to the HPLMN and the File Transfer service is authorised, the client shall (re-)register in IMS to add the media

feature tag for File Transfer via HTTP.
If the value of the configuration parameter transits from 1 to 0 or from 2 to 0 while the device is connected to a cellular access network other than the HPLMN and File Transfer via HTTP is registered in IMS, the client shall de- or re-register in IMS to remove the media feature tag for File Transfer via HTTP.

- Associated HTTP XML characteristic type: “fileTransferDataOff”

Node: /<x>/Services/Ext/DataOff/smsoIPDataOff

Controls the SMS over IP service behaviour when the cellular data switch is switched off.

The parameter is only applicable in case the SMS over IP service is supported. It will not be instantiated otherwise.

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

Table 114: Data Off Services Extension MO sub tree addition parameters (smsoIPDataOff)

- Values:
  0: the SMS over IP service is not available when cellular data is switched off.
  1: the SMS over IP service is available when cellular data is switched off
  2: the SMS over IP service is available on cellular networks when cellular data is switched off and the device is attached to the HPLMN, otherwise the service is disabled.
- Post-reconfiguration actions:
- If the value of the configuration parameter transits from 0 to 1 or from 0 to 2 while the device is connected to LTE in the HPLMN and the SMS over IP service is authorised, the client shall (re-)register in IMS to add the media feature tag for SMS over IP.
  If the value of the configuration parameter transits from 1 to 0 or from 2 to 0 while the device is connected to a cellular access network other than the HPLMN and SMS over IP is registered in IMS, the client shall de- or re-register in IMS to remove the media feature tag for SOS over IP. The client shall enable SMS over circuit switched access. SMS over GPRS or SMS over SGs as applicable.
- Associated HTTP XML characteristic type: “smsoIPDataOff”

Node: /<x>/Services/Ext/DataOff/mmsDataOff

Controls the MMS behaviour when the cellular data switch is switched off.

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

Table 115: Data Off Services Extension MO sub tree addition parameters (mmsDataOff)

- Values:
  0: the MMS is not available when cellular data is switched off.
  1: the MMS is available when cellular data is switched off

2: the MMS is available when cellular data is switched off and the client is connected to a cellular access network other than the HPLMN.

- Post-reconfiguration actions: no specific actions.
- Associated HTTP XML characteristic type: “mmsDataOff”

Node: /<x>/Services/Ext/DataOff/contentShareDataOff

Controls the Video Share and Image Share service behaviour when the cellular data switch is switched off.

The parameter is only applicable in case the Video Share and/or Image Share services are supported. It will not be instantiated otherwise.

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

Table 116: Data Off Services Extension MO sub tree addition parameters (contentShareDataOff)

- Values:
  0: the Video Share, Image Share, Shared Map and Shared Sketch services are not available on cellular networks when cellular data is switched off
  1: the Video Share, Image Share, Shared Map and Shared Sketch services are available on cellular networks when cellular data is switched off
  2: the Video Share, Image Share, Shared Map and Shared Sketch services are not available on cellular networks when cellular data is switched off and the device is not attached to the HPLMN.
- Post-reconfiguration actions: If the value of the configuration parameter transits from 0 to 1 or from 0 to 2 while the device is connected to the HPLMN and at least one of Image Share, Video Share, Shared Map or Shared Sketch is authorised the client shall (re-)register in IMS to add the relevant media feature tags according to the authorisation of these services.
  If the value of the configuration parameter transits from 1 to 0 or from 2 to 0 while the device is connected to a cellular access network other than the HPLMN and at least one of the Video Share, Image Share, Shared Map or Shared Sketch service is registered in IMS, the client shall de-or re-register with in IMS to remove the media feature tags for the service being disabled by the configuration parameter.
- Associated HTTP XML characteristic type: “contentShareDataOff”

Node: /<x>/Services/Ext/DataOff/preAndPostCallDataOff

Controls the Call Composer and Post-call service behaviour when the cellular is switched off.

The parameter is only applicable in case the Call Composer and/or Post-call services are supported. It will not be instantiated otherwise.

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

Table 117: Data Off Services Extension MO sub tree addition parameters (preAndPostCallDataOff)

- Values:
  0: the Call Composer and Post-call services are not available on cellular networks when cellular data is switched off
  1: the Call Composer and Post-call services are available on cellular networks when cellular data is switched off
  2: the Call Composer and Post-call services are not available when cellular data is switched off and the device is attached to a cellular network other than the HPLMN..
- Post-reconfiguration actions: If the value of the configuration parameter transits from 0 to 1 or from 0 to 2 while the device is connected to the HPLMN and at least one of Call Composer and Post-call is authorised the client shall (re-)register in IMS to add the relevant media feature tags according to the authorisation of these services.
  If the value of the configuration parameter transits from 1 to 0 or from 2 to 0 while the device is connected to a cellular access network other than the HPLMN and at least one of Call Composer and Post-call service is registered in IMS, the client shall de- or re-register with in IMS to remove the media feature tags for the service being disabled by the configuration parameter.
- Associated HTTP XML characteristic type: “preAndPostCallDataOff”

<u>Node: /<x>/Services/Ext/DataOff/volteDataOff</u>

Controls the VoLTE service behaviour when the cellular data switch is switched off.

The parameter is only applicable in case the VoLTE services are supported. It will not be instantiated otherwise.

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

Table 118: Data Off Services Extension MO sub tree addition parameters (volteDataOff)

- Values:
  0: the VoLTE service is not available when cellular data is switched off.
  1: the VoLTE service is available when cellular data is switched off
  2: the VoLTE service is not available when cellular data is switched off and the device is attached to a cellular network other than the HPLMN.
- Post-reconfiguration actions: If the value of the configuration parameter transits from 0 to 1 or from 0 to 2 while the device is connected to LTE in the HPLMN, the client shall (re-)register in IMS to add the media feature tag for Multimedia Telephony.
  If the value of the configuration parameter transits from 1 to 0 or from 2 to 0 while the device is connected to a LTE access network other than the HPLMN, the client shall de- or re-register in IMS to remove the media feature tag for Multimedia Telephony.
- Associated HTTP XML characteristic type: “volteDataOff”

<u>Node: /<x>/Services/Ext/DataOff/IPVideoCallDataOff</u>

Controls the IP Video Calling service behaviour when the cellular data switch is switched off.

The parameter is only applicable in case the RCS IP Video Calling or IR94 Video Calling service is supported on cellular networks. It will not be instantiated otherwise.

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

Table 119: Data Off Services Extension MO sub tree addition parameters (IPVideoCallDataOff)

- Values:
  0: the IP Video Calling service is not available on cellular networks when cellular data is switched off.
  1: the IP Video Calling service is available when cellular data is switched off
  2: the IP Video Calling service is not available when cellular data is switched off and the device is connected to a cellular network other than the HPLMN.
- Post-reconfiguration actions: If the value of the configuration parameter transits from 0 to 1 or from 0 to 2 while the device is connected to the HPLMN and the IP Video Calling service is authorised, the client shall (re-)register in IMS to add the media feature tag for IP Video Calling.
  If the value of the configuration parameter transits from 1 to 0 or from 2 to 0 while the device is connected to a cellular access network other than the HPLMN and IP Video Calling service is registered in IMS, the client shall de- or re-register in IMS to remove the media feature tag for IP Video Calling.
- Associated HTTP XML characteristic type: “IPVideoCallDataOff”

Node: /<x>/Services/Ext/DataOff/provisioningDataOff

Controls the device/client behaviour related to provisioning requests when the cellular data switch is switched off.

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

Table 120: Data Off Services Extension MO sub tree addition parameters (provisioningDataOff)

- Values:
  0: No provisioning requests shall be sent over cellular networks when cellular data is switched off.
  1: provisioning requests may be sent over cellular networks when cellular data is switched off
  2: no provisioning requests shall be sent over cellular networks other than the HPMN when cellular data is switched off.
- Post-reconfiguration actions: No specific actions.
- Associated HTTP XML characteristic type: “provisioningDataOff”

Node: /<x>/Services/Ext/DataOff/extensionsDataOff

Controls the extension to extension service (see section 3.12.2.2) behaviour when the cellular data switch is switched off.

The parameter is only applicable in case the extension to extension service is supported. It will not be instantiated otherwise.

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

Table 121: Data Off Services Extension MO sub tree addition parameters (extensionsDataOff)

- Values:
  0: the Extension to Extension service is not available on cellular networks when cellular data is switched off.
  1: the Extension to Extension service is available when cellular data is switched off
  2: the Extension to Extension service is not available when cellular data is switched off and the device is connected to a cellular network other than the HPLMN.
- Post-reconfiguration actions: If the value of the configuration parameter transits from 0 to 1 or from 0 to 2 while the device is connected to the HPLMN and Extension services service are authorised, the client shall (re-)register in IMS to add the media feature tag for Extension services.
  If the value of the configuration parameter transits from 1 to 0 or from 2 to 0 while the device is connected to a cellular access network other than the HPLMN and Extension services are registered in IMS, the client shall de- or re-register in IMS to remove all media feature tags for Extension services.
- Associated HTTP XML characteristic type: “extensionsDataOff”

Node: /<x>/Services/Ext/DataOff/syncDataOff

Controls the behaviour for synchronisation with the Common Message when the cellular data is switched off.

The parameter is only applicable in case the Common Message Store is supported. It will not be instantiated otherwise.

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

Table 122: Data Off Services Extension MO sub tree addition parameters (syncDataOff)

- Values:
  0: the synchronisation with the Common Message Store is not available on cellular networks when cellular data is switched off
  1: the synchronisation with the Common Message Store is available on cellular networks when cellular data is switched off
  2: the synchronisation with the Common Message Store is available on cellular networks when cellular data is switched off if the device is attached to the HPLMN, otherwise the services are disabled.

- Post-reconfiguration actions: If the value of the configuration parameter transits from 1 to 0 or from 2 to 0 while the device is connected to a cellular access network other than the HPLMN the client should trigger a data connection triggered synchronization with the Common Message Store.
- Associated HTTP XML characteristic type: “syncDataOff”

A.2.2. IMS sub tree additions

Additions to the IMS MO defined in [3GPP TS 24.167] have been specified in [PRD-RCC.15].This section expands the management object defined in that specification with some additional RCS specific parameters that are added to the existing IMS extension tree used in [PRD-RCC.15] (i.e. the node <x> is the same node as used for the extensions defined in [PRD-RCC.15]). Following parameters have been defined:

Node: <x>/rcsVolteSingleRegistration

Where <x> corresponds to the Ext node of the IMS sub tree defined in [3GPP TS 24.167].

Leaf node that describes the behaviour regarding the instantiation of the IMS stack in devices supporting both RCS and VoLTE. It is also used to control the APN selection on devices that are not enabled for VoLTE as described in section 2.9.1.4.

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	int	Get, Replace

Table 123: IMS Configuration sub tree addition parameters (rcsVolteSingleRegistration)

- Values:
  - 0, the device shall follow a dual registration approach (transition solution) where RCS services use a separate registration from the VoLTE/ViLTE one.
  - 1 (default if not provided), the device shall follow a single registration (target solution) for both RCS and VoLTE/ViLTE services.
  - 2, the device shall follow a single registration for both RCS and VoLTE/ViLTE services when in the home network, and shall follow a dual registration when roaming (transition solution).
- 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: “rcsVolteSingleRegistration”

The resulting HTTP configuration XML structure is presented in the table below

```
<characteristic type="IMS">
    …
    <characteristic type="Ext">
      …
      <parm name=" rcsVolteSingleRegistration" value="X"/>
    <characteristic type="Ext"/>
</characteristic>
```

Table 124: IMS sub tree associated HTTP configuration XML structure

A.2.3. Presence sub tree additions

RCS includes the following additions to the Presence MO sub tree where <*Presence*> corresponds to the <*x*> root node of the Presence MO defined in [PRESENCE2MO].

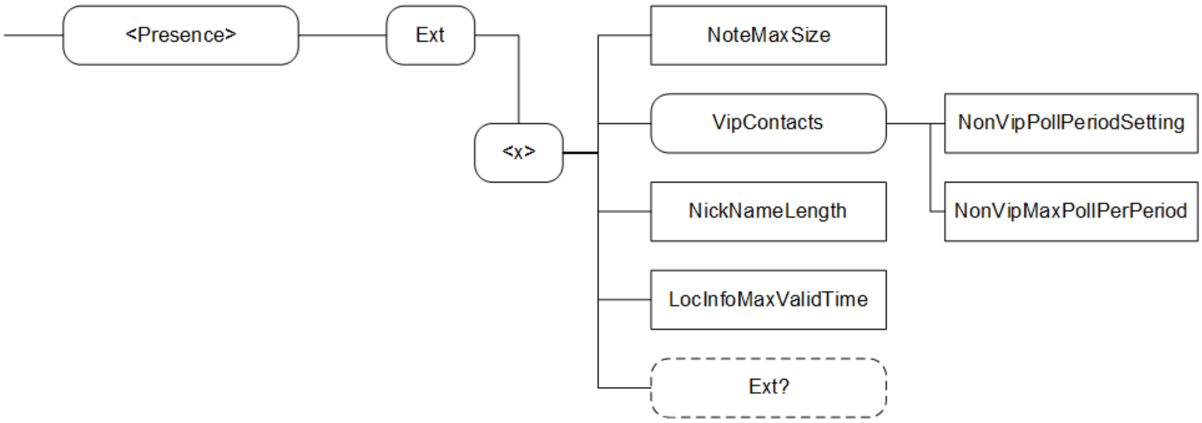

Figure 121: RCS additions to the Presence MO sub tree

The associated HTTP configuration XML structure associated with the Presence parameters (both from the Presence MO defined in [PRESENCE2MO] and the RCS specific parameters (shown in blue)) is presented in the table below

```
<characteristic type="PRESENCE">

        <parm name="NoteMaxSize" value="X"/>
        <characteristic type="VIPCONTACTS">
              <parm name="NonVipPollPeriodSetting" value="X"/>
              <parm name="NonVipMaxPollPerPeriod" value="X"/>
        </characteristic>
        <parm name="NickNameLength" value="X"/>
        <parm name="LocInfoMaxValidTime" value="X"/>
        <characteristic type="Ext"/>
        <parm name="client-obj-datalimit" value="X"/>
        <parm name="content-serveruri" value="X"/>
        <parm name="source-throttlepublish" value="X"/>
        <parm name="max-number-ofsubscriptions-inpresence-list" value="X"/>
        <parm name="service-uritemplate" value="X"/>
        <parm name="RLS-URI" value="X"/>
</characteristic>
```

Table 125 : Presence sub tree associated HTTP configuration XML structure

<u>Node: <x></u>

Under this interior node the RCS parameters related to Presence are placed

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 126: Presence MO sub tree addition presence node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-Presence:6.0*
- Associated HTTP XML characteristic type: “PRESENCE”

Node: <x>/NoteMaxSize

Leaf node that represent the maximum authorised size for a note

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

Table 127: Presence MO sub tree addition parameters (NoteMaxSize)

- Values: < Note maximum length in characters>

NOTE: This should be set to a value that is lower than the one defined at watcher side in the OMA Presence Implementation guideline [PRESENCEIG].

- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “NoteMaxSize”

Node: <x>/NickNameLength

Leaf node that represents the maximum number of characters allowed for the user chosen nickname.

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

Table 128: Presence MO sub tree addition parameters (NickNameLength)

- Values: must be less or equal to 200

NOTE: An RCS client must be able to handle of up to 200 characters

- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “NickNameLength”

Node: <x>/LocInfoMaxValidTime

Leaf node that represents the maximum validity duration time for a location item.

This parameter must be taken account by the device presence UA when setting the “until” attribute of the presence items place-type, time-offset and the usage-rule/retention-expiry item value

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

Table 129: Presence MO sub tree addition parameters (LocInfoMaxValidTime)

- Values: < Validity time in seconds>, when set to 0 there is no limit to the validity time
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “LocInfoMaxValidTime”

Node: <x>/VipContacts

Interior node where VIP contacts related parameters are stored

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 130: Presence MO sub tree addition VIP Contacts node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-Presence:6.0:VipContacts*
- Associated HTTP XML characteristic type: “VIPCONTACTS”

Node: <x>/VipContacts/NonVipPollPeriodSetting

Leaf node that indicates, in seconds, the period duration for the calculation of the number of Poll operations on the non-VIP Contacts (“rcs_poll”) RLS list authorised during this period

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

Table 131: Presence MO sub tree addition parameters (NonVipPollPeriodSetting)

- Values: integer that represents a time value in seconds
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “NonVipPollPeriodSetting”

Node: <x>/VipContacts/NonVipMaxPollPerPeriod

Leaf node that indicates the maximum number of Poll operations on the non-VIP Contacts (“rcs_poll”) RLS list that are authorised for the User Agent during each period (period parameter defined in the previous /VipContacts/NonVipPollPeriodSetting node).

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

Table 132: Presence MO sub tree addition parameters (NonVipMaxPollPerPeriod)

- Values: integer that represents the total amount of Poll operations on the non-VIP Contacts list per each period.

- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “NonVipMaxPollPerPeriod”

Node: <x>/Ext

An extension node for Service Provider specific parameters. Clients that are not aware of any extensions in this subtree (e.g. because they are not Service Provider specific) should not instantiate this tree.

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

Table 133: Presence MO sub tree addition Service Provider Extension Node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-Presence:6.0:Ext*
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML characteristic type: “Ext”

A.2.4. XDMS sub tree additions

RCS includes the following additions to the XDMS MO sub tree where *<XDMS>* corresponds to the *<x>* root node of the XDMS MO defined in [XDMMO].

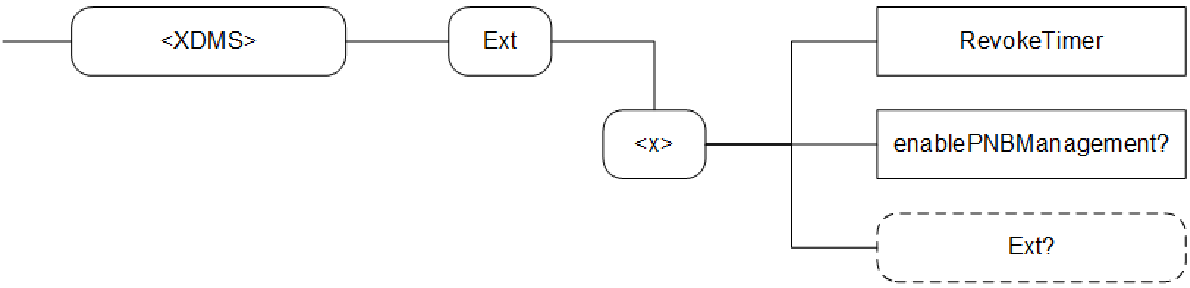

Figure 122: RCS additions to the XDMS MO sub tree

The associated HTTP configuration XML structure associated with the XDMS parameters (both from the XDMS MO defined in [XDMMO] and the RCS specific parameters (shown in blue)) is presented in the table below

```
<characteristic type=”XDMS”>
        <parm name=”RevokeTimer” value=”X”/>
        <parm name=”enablePNBManagement” value=”X”/>
        <characteristic type=”Ext”/>
        <parm name=”XCAPRootURI” value=”X”/>
        <parm name=”XCAPAuthenticationUserName” value=”X”/>
        <parm name=”XCAPAuthenticationSecret” value=”X”/>
        <parm name=”XCAPAuthenticationType” value=”X”/>
</characteristic>
```

Table 134 : XDMS sub tree associated HTTP configuration XML structure

Node: <x>

Under this interior node the RCS parameters related to XDM are placed

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 135: XDM MO sub tree addition xdm node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-xdm:6.0*
- Associated HTTP XML characteristic type: “XDMS”

Node: <x>/RevokeTimer

Leaf node that indicates the duration a contact should remain in the RCS revocation list. It may also be used for the frequency that the list is checked.

It is instantiated only if Social Presence is enabled.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

Table 136: XDMS MO sub tree addition parameters (RevokeTimer)

- Values: < Timer value in seconds>
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “RevokeTimer”

Node: <x>/enablePNBManagement

Leaf node that describes whether the PNB feature is turned ON or OFF

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

Table 137: XDMS MO sub tree addition parameters (enablePNBManagement)

- Values:
  0 or not instantiated, the PNB feature is not used.
  1, the PNB feature is ON, the PNB lists can be managed by the user and applied by BPEF
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “enablePNBManagement”

<u>Node: <x>/Ext</u>

An extension node for Service Provider specific parameters. Clients that are not aware of any extensions in this subtree (e.g. because they are not Service Provider specific) should not instantiate this tree.

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

Table 138: XDMS MO sub tree addition Service Provider Extension Node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-xdm:6.0:Ext*
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML characteristic type: “Ext”

A.2.5. SUPL sub tree additions

RCS includes the following additions to the SUPL MO sub tree where <*SUPL*> corresponds to the <*x*> root node of the SUPL MO defined in [SUPLMO]:

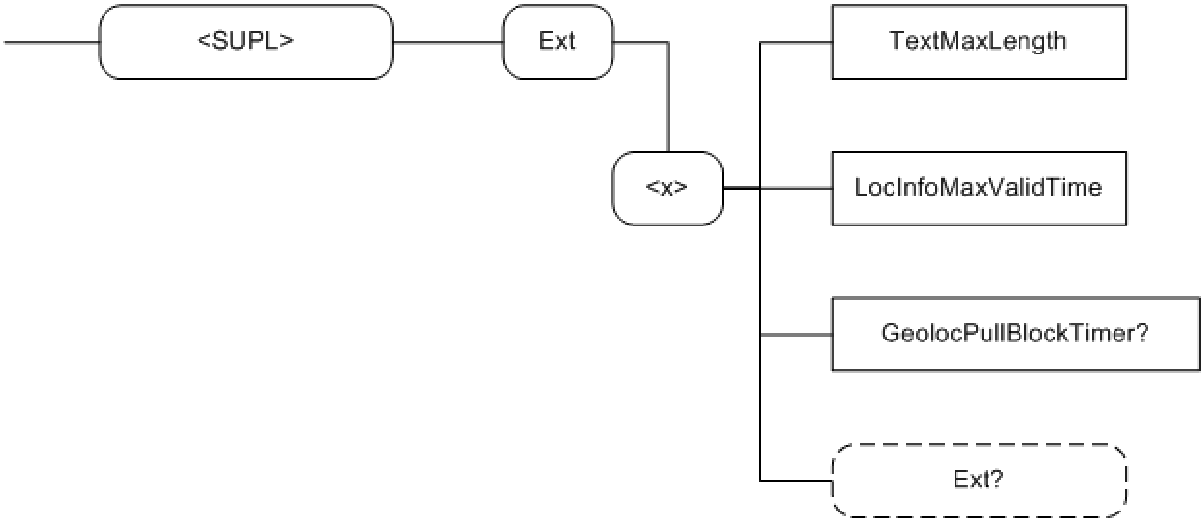

Figure 123 : RCS additions to the SUPL MO sub tree

The associated HTTP configuration XML structure associated to the geolocation parameters (both from the SUPL MO defined in [SUPLMO] and the RCS specific parameters (shown in blue)) is presented in the table below

```
<characteristic type=”SUPL”>
        <parm name=”TextMaxLength” value=”X”/>
        <parm name=”LocInfoMaxValidTime” value=”X”/>
        <parm name=”geolocPullBlockTimer” value=”X”/>
        <characteristic type=”Ext”/>
</characteristic>
```

Table 139 : SUPL sub tree associated HTTP configuration XML structure

Node: <x>

Under this interior node the RCS parameters related to the geolocation configuration are placed.

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 140: SUPL MO sub tree addition geoloc node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-supl:6.0*
- Associated HTTP XML characteristic type: “SUPL”

Node: <x>/TextMaxLength

Leaf node that represents the maximum numbers of characters authorised for the textual attribute of the location information provided in the geolocation PUSH and PULL services

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

Table 141: SUPL MO sub tree addition parameters (TextMaxLength)

- Values: must be less or equal to 200.
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “TextMaxLength”

Node: <x>/LocInfoMaxValidTime

Leaf node that represents the maximum validity duration time for a location item

This parameter must be taken account by the device providing the location information when setting the “until” attribute of the items time-offset and the usage-rule/retention-expiry item value

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

Table 142: SUPL MO sub tree addition parameters (LocInfoMaxValidTime)

- Values: < Validity time in seconds>
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “LocInfoMaxValidTime”

Node: <x>/GeolocPullBlockTimer

Leaf node that represents the interval during which the Geolocation PULL application is not allowed to send a PULL request to a target contact if a previous request was explicitly rejected. The parameter is only applicable if the Geolocation PULL service is based on File Transfer technology and will thus not be instantiated in case that is not supported.

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

Table 143: SUPL MO sub tree addition parameters (GeolocPullBlockTimer)

- Values: <Timer value in seconds>
  The value represents the duration, in case of explicit operation denied by a target contact, during which a new Pull operation in not allowed to be initiated.
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “geolocPullBlockTimer”

Node: <x>/Ext

An extension node for Service Provider specific parameters. Clients that are not aware of any extensions in this subtree (e.g. because they are not Service Provider specific) should not instantiate this tree.

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

Table 144: SUPL MO sub tree addition Service Provider Extension Node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-supl:6.0:Ext*
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML characteristic type: “Ext”

A.2.6. IM sub tree additions

RCS includes the following additions to the IM MO sub tree where *<IM>* corresponds to the *<x>* root node of the IM MO described in [RCS-SIMPLEIM-ENDORS]:

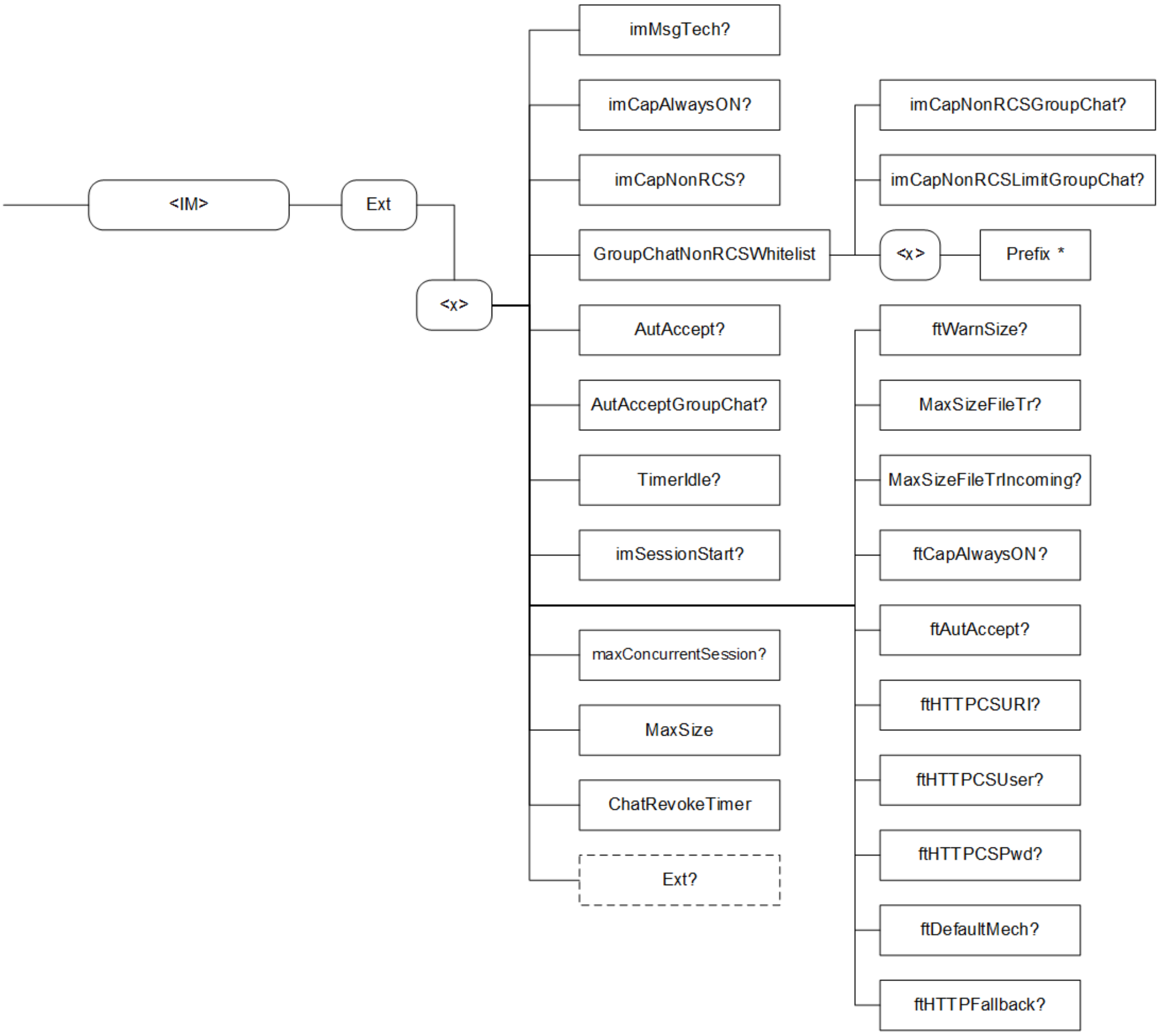

Figure 124: RCS additions to the IM MO sub tree

The associated HTTP configuration XML structure associated to the IM parameters (both from the IM MO defined in [RCS-SIMPLEIM-ENDORS] and the RCS specific parameters (shown in blue)) is presented in the table below

```
<characteristic type="IM">
        <parm name="imMsgTech" value="X"/>
        <parm name="imCapAlwaysON" value="X"/>
        <parm name="imCapNonRCS" value="X"/>
        <characteristic type="GroupChatNonRCSWhitelist" value="X"/>
              <parm name="imCapNonRCSGroupChat" value="X"/>
              <parm name="imCapNonRCSLimitGroupChat" value="X"/>
              <characteristic type="GroupChatAllowedPrefixes">
                    <parm name=" Prefix1" value="X"/>
                    <parm name=" Prefix2" value="X"/>
                    <parm name=" Prefix3" value="X"/>
                    …
              </characteristic>
        </characteristic>
        <parm name="AutAccept" value="X"/>
        <parm name="AutAcceptGroupChat" value="X"/>
        <parm name="imSessionStart" value="X"/>
        <parm name="TimerIdle" value="X"/>
        <parm name="MaxSize" value="X"/>
        <parm name="ChatRevokeTimer" value="X"/>
        <parm name="ftWarnSize" value="X"/>
        <parm name="MaxSizeFileTr" value="X"/>
        <parm name="MaxSizeFileTrIncoming" value="X"/>
        <parm name="ftCapAlwaysON" value="X"/>
        <parm name="ftAutAccept" value="X"/>
        <parm name="ftHTTPCSURI" value="X"/>
        <parm name="ftHTTPCSUser" value="X"/>
        <parm name="ftHTTPCSPwd" value="X"/>
        <parm name="ftDefaultMech" value="X"/>
        <parm name="ftHTTPFallback" value="X"/>
        <characteristic type="Ext"/>
        <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 145 : IM sub tree associated HTTP configuration XML structure

Node: <x>

Under this interior node the RCS parameters related to the IM configuration are placed.

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 146: IM MO sub tree addition IM node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-im:6.0*
- Associated HTTP XML characteristic type: “IM”

Node: <x>/imMsgTech

Leaf node that describes parameter allows selecting what technology is used for the chat service described in sections 3.3 and 3.4 as well as for the File Transfer service in section 3.5.

It is required to be instantiated if a service provider enables Chat.

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

Table 147: IM MO sub tree addition parameters (imMsgTech)

- Values: 1, CPM as specified in [RCS-CPM-CONVFUNC-ENDORS]. 0 (default if not provided), SIMPLE IM as specified in [RCS-SIMPLEIM-ENDORS].
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML parameter ID: “imMsgTech”

Node: <x>/imCapAlwaysON

Leaf node that describes whether the Chat capability needs to be on independently of whether or not the other end is available in IMS.

It is required to be instantiated if a service provider enables Chat.

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

Table 148: IM MO sub tree addition parameters (IMCAPAlwaysOn)

- Values:
  1, it is enabled, Chat capability for all RCS contacts is available independently of whether the other end is available in IMS;
  0, it is disabled and thus Chat capability is reported based on the algorithm in section 2.7.1.1. When a service provider prefers that the client does not use Chat when an RCS contact is not currently registered in IMS, it sets the parameter to 0
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “imCapAlwaysOn”

Node: <x>/imCapNonRCS

Leaf node that describes whether the Chat capability needs to be on independently of whether or not the other end is an RCS contact. For example this can be used in Service Providers providing the interworking functionality for Chat

It is required to be instantiated if a service provider enables Chat.

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

Table 149: IM MO sub tree addition parameters (IMCAPNonRCS)

- Values: 1, RCS Messaging Server based interworking is enabled; 0, it is disabled
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “imCapNonRCS”

Node: <x>/GroupChatNonRCSWhitelist

A Placeholder interior node for the configuration related to a whitelist for non RCS users invited to a group chat

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 150: IM MO sub tree GroupChatNonRCSWhitelist node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-im:6.0:gcnonrcswhitelist*
- Associated HTTP XML parameter ID: “GroupChatNonRCSWhitelist”

Node: <x>/GroupChatNonRCSWhitelist/imCapNonRCSGroupChat

Leaf node that describes whether and under which conditions the device is able to add non RCS contacts in a Group Chat. For example this can be used by Service Providers providing the interworking functionality for Group Chat.

It is required to be instantiated if a service provider enables Group Chat.

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

Table 151: IM MO sub tree addition parameters (IMCAPNonRCSGroupChat)

- Values: This parameter can have 3 possible values:
  0 - The device is not able to add any non RCS contact in any Group Chat session
  1 - The device is able to add non RCS contacts in any Group Chat session
  2 - The device is able to add non RCS contacts only upon Group Chat creation and only in Group Chat sessions generated by its user
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “imCapNonRCSGroupChat”

Node: <x>/GroupChatNonRCSWhitelist/imCapNonRCSLimitGroupChat

Leaf node that describes the number of non RCS contacts that can be added by the user that creates a Group Chat upon Group Chat generation. For example this can be used by Service Providers providing the interworking functionality for Group Chat.

It is required to be instantiated if a service provider enables Group Chat.

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

Table 152: IM MO sub tree addition parameters (IMCAPNonRCSLimitGroupChat)

- Values: <max number of non RCS users added in a Group Chat upon Group Chat generation by the user that creates the Group Chat>, when set to 0 this limit does not apply
- 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 (see section 2.4) using the new parameter
- Associated HTTP XML parameter ID: “imCapNonRCSLimitGroupChat”

Node: <x>/GroupChatNonRCSWhitelist/<x>

A Placeholder interior node where to place 1 or more Prefix leaf nodes

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 153: IM MO sub tree GroupChatAllowedPrefixes node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-im:6.0:gcnonrcswhitelist:prefixes*
- Associated HTTP XML characteristic type: “GroupChatAllowedPrefixes”

Node: <x>/GroupChatNonRCSWhitelist/<x>/Prefix

Leaf node that represents a prefix configured by the Service Provider

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

Table 154: IM MO sub tree addition parameters (Prefix)

- Values: <a Service Provider defined prefix>
  The prefix is used to identify phone numbers of non RCS contacts allowed to be added to a Group Chat. For the purpose of matching a phone number with the prefix the client shall remove visual separators and white space from the input phone number string. The phone numbers are identified by matching with the prefix value starting from the left. Its length can be one or more digits and it can start with the "+" character.
  Examples: +446, +4479, 00446, 004479, 06, 079
  In addition the client shall support matching of phone numbers based on regular expression. To indicate that the Prefix parameter value contains a regular expression it shall start with the "!" character. The subsequent string shall be interpreted by the

client to match the phone number string using Portable Operating System Interface (POSIX) extended regular expression (see [POSIX]).
Example: !^(\+44|0044|0)(6|79)\d*

- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “Prefix<X>” where <X> is a positive integer value

Node: <x>/AutAccept

Leaf node that represent the automatic/manual chat session answer mode

It is required to be instantiated if a service provider enables 1-to-1 Chat.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	bool	Get

Table 155: IM MO sub tree addition parameters (AutAccept)

- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “AutAccept”

Node: <x>/imSessionStart

Leaf node that describes when the receiver client/device implementation should return the 200 OK initiating the MSRP session associated to a 1-to-1 chat. Please note that this parameter is transparent to the user.

It is required to be instantiated if a service provider enables 1-to-1 Chat.

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

Table 156: IM MO sub tree addition parameters (imSessionStart)

- Values: This parameter can have 3 possible values:

  - 0 (RCS 6 default):
    The 200 OK is sent when the receiver consumes the notification by opening the chat window.
  - 1 (RCS Release 2-4 default):
    The 200 OK is sent when the receiver starts to type a message to be sent back in the chat window.
  - 2 (new option):
    The 200 OK is sent when the receiver presses the button to send a message

(That is the message will be buffered in the client until the MSRP session is established).

- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “imSessionStart”

Node: <x>/AutAcceptGroupChat

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

It is required to be instantiated if a service provider enables Group Chat.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	bool	Get

Table 157: 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “AutAcceptGroupChat”

Node: <x>/TimerIdle

Leaf node that represents the timeout for a chat session in idle mode (when there is no chat user activity)

It is required to be instantiated if a service provider enables Chat.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

Table 158: IM MO sub tree addition parameters (TimerIdle)

- Values: <Timer value in seconds>
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “TimerIdle”

Node: <x>/MaxSize

Leaf node that represent the maximum authorised size of the content payload of a chat message in a Chat session without transfer encoding. The parameter is applicable for Chat messages transferred either via SIP INVITE or MSRP requests.

It is required to be instantiated if a service provider enables 1-to-1 or Group Chat.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

Table 159: IM MO sub tree addition parameters (MaxSize)

- Values: <content maximum size in bytes>
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “MaxSize”

Node: <x>/ChatRevokeTimer

Leaf node that represents the time the service provider allows to elapse after the client has sent the message and before Revoke Message request is automatically triggered by the client when it has not received the delivery notification for that message.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

Table 160: IM MO sub tree addition parameters (ChatRevokeTimer)

- Values: <Timer value in seconds>
  When set to 0, the client is not able to send MessageRevoke requests
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “ChatRevokeTimer”

Node: <x>/ftWarnSize

Leaf node that describes the file transfer size threshold (in KB) when the user should be warned about the potential charges associated to the transfer of a large file.

It is required to be instantiated if a service provider enables File Transfer.

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

Table 161: IM MO sub tree addition parameters (ftWarnSize)

- Values: The file size threshold (in KB) or 0 to disable the warning
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “ftWarnSize”

Node: <x>/MaxSizeFileTr

Leaf node that represent the maximum authorised size of a file that can be transfers using the RCS File Transfer service

It is required to be instantiated in case a service provider enables File Transfer.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

Table 162: IM MO sub tree addition parameters (MaxSizeFileTr)

- Values: The maximum file size threshold (in KB) or 0 to disable the limit
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “MaxSizeFileTr”

Node: <x>/MaxSizeFileTrIncoming

Leaf node that represent the maximum authorised size of a file that can be received using the RCS File Transfer service

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

Table 163: IM MO sub tree addition parameters (MaxSizeFileTrIncoming)

- Values: The maximum file size threshold (in KB) or 0 to disable the limit
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “MaxSizeFileTrIncoming”

Node: <x>/ftCapAlwaysON

Leaf node that describes whether the file transfer capability needs to be on independently of whether or not the other end is registered

It is required to be instantiated if a service provider enables File Transfer.

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

Table 164: IM MO sub tree addition parameters (ftCapAlwaysOn)

- Values: 1, RCS Messaging Server based store and forward is enabled; 0, it is disabled. It shall be taken into account that this parameter can be only set to 1 if:

    - All the interconnected service providers support the file transfer store and forward feature, or,
    - store and forward for files is provided as an originating function

- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “ftCapAlwaysOn”

Node: <x>/ftAutAccept

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

It is required to be instantiated if a service provider enables File Transfer.

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

Table 165: 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 File Transfer warning size as configured by the FT WARN SIZE 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “ftAutAccept”

Node: <x>/ftHTTPCSURI

This parameter configures the URI of the HTTP content server where files are going to be uploaded on the originating side if the destination cannot accept within the validity period.

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

Table 166: IM MO sub tree addition parameters (ftHTTPCSURI)

- Values: The string containing the URI of the HTTP content server in charge of storing the files.
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “ftHTTPCSURI”

Node: <x>/ftHTTPCSUser

This parameter is the value of the user value that shall be used to authenticate the RCS client trying to either get a root URL (HTTP GET request) or upload a file (HTTP post request).

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	chr	No Get, No Copy

Table 167: IM MO sub tree addition parameters (ftHTTPCSUser)

- Values: The string containing **user 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 (see section 2.4) using the new parameter.

- Associated HTTP XML parameter ID: "ftHTTPCSUser"

Node: <x>/ftHTTPCSPwd

This parameter is the value of the password value that shall be used to authenticate the RCS client trying to either get a root URL (HTTP GET request) or upload a file (HTTP post request).

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	chr	No Get, No Copy

Table 168: IM MO sub tree addition parameters (ftHTTPCSPwd)

- Values: The string containing **password 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: "ftHTTPCSPwd"

Node: <x>/ftDefaultMech

Leaf node that describes which file transfer mechanism (MSRP or HTTP) shall be used by default if both ends support both of them

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

Table 169: IM MO sub tree addition parameters (ftDefaultMech)

- Values:
  - MSRP
  - HTTP
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: "ftDefaultMech"

Node: <x>/ftHTTPFallback

Leaf node that controls how multimedia content and files are sent to contacts that do not support any of the enabled File Transfer mechanisms (including non RCS contacts)

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

Table 170: IM MO sub tree addition parameters (ftHTTPFallback)

- Values:
  - 0, MMS is used
  - 1, The mechanism described in section 3.5.4.8.6 is used

- Post-reconfiguration actions: the configured way to transfer the content is used for any future transfers to contacts that do not support any of the enabled File Transfer mechanisms.
- Associated HTTP XML parameter ID: “ftHTTPFallback”

Node: <x>/Ext

An extension node for Service Provider specific parameters. Clients that are not aware of any extensions in this subtree (e.g. because they are not Service Provider specific) should not instantiate this tree.

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

Table 171: IM MO sub tree addition Service Provider Extension Node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-im:6.0:Ext*
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML characteristic type: “Ext”

A.2.7. CPM MO sub tree

RCS includes the following additions as a new configuration sub tree, the CPM MO subtree

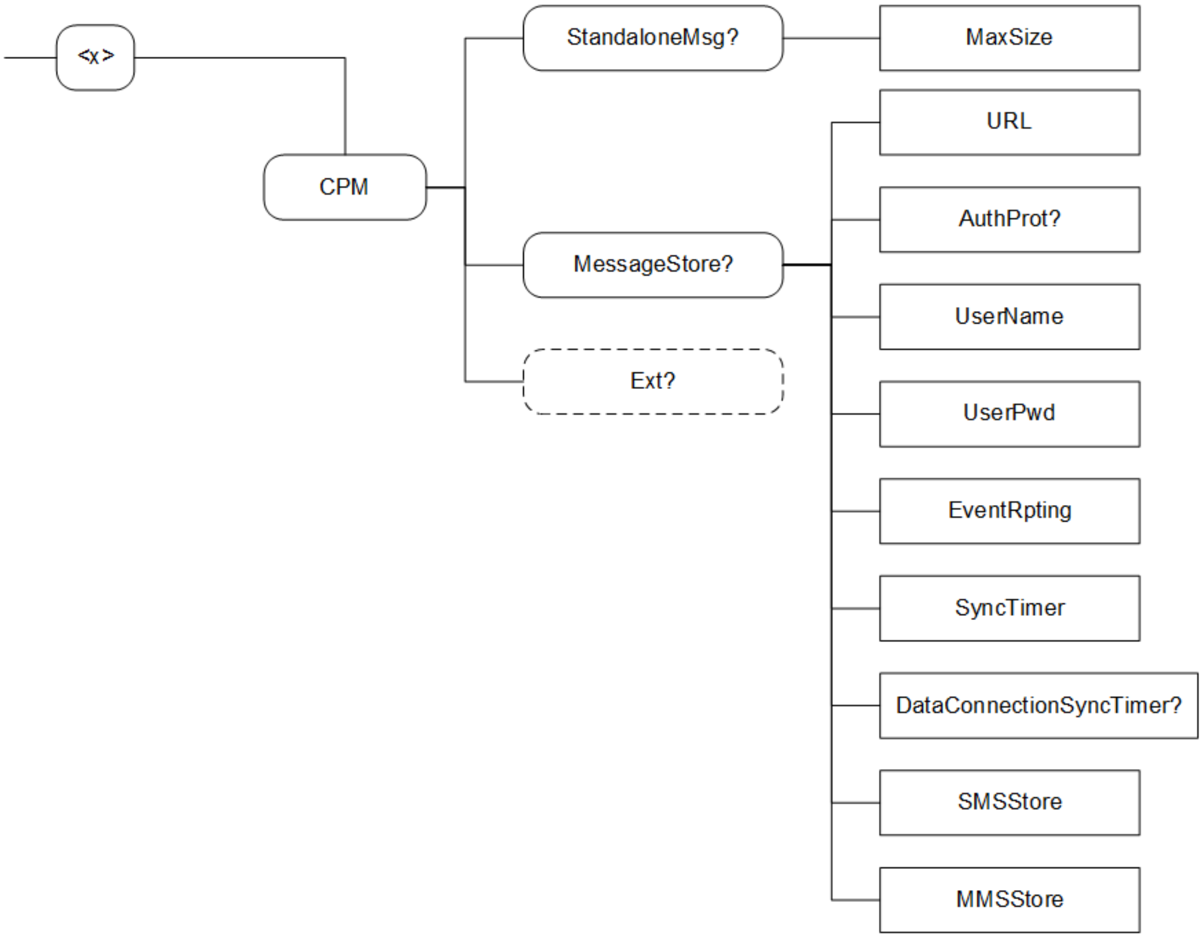

Figure 125: RCS additions, CPM MO sub tree

The associated HTTP configuration XML structure associated to the CPM parameters is presented in the table below. Only RCS specific parameters (shown in blue) are included as OMA does not define a CPM MO.

```
<characteristic type="CPM">
    <characteristic type="StandaloneMsg">
        <parm name="MaxSizeStandalone" value="X"/>
    </characteristic>
    <characteristic type="MessageStore">
        <parm name="Url" value="X"/>
        <parm name="AuthProt" value="X"/>
        <parm name="UserName" value="X"/>
        <parm name="UserPwd" value="X"/>
        <parm name="EventRpting" value="X"/>
        <parm name="AuthArchive" value="X"/>
        <parm name="SyncTimer" value="X"/>
        <parm name="DataConnectionSyncTimer" value="X"/>
        <parm name="SMSStore" value="X"/>
        <parm name="MMSStore" value="X"/>
    </characteristic>
    <characteristic type="Ext"/>
</characteristic>
```

Table 172 : CPM sub tree associated HTTP configuration XML structure

Node: /<x>/CPM

Under this interior node the RCS parameters related to the CPM configuration are placed.

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 173: CPM MO sub tree addition CPM node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-cpm:6.0*
- Associated HTTP XML characteristic type: “CPM”

Node: /<x>/CPM/StandaloneMsg

Interior node where are filled parameters related to the RCS Text message and Multimedia message service

This node is not instantiated if the Service Provider does not enable Standalone Messaging.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	node	Get

Table 174: CPM MO sub tree addition Standalone messaging node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-cpm:6.0:StandaloneMsg*
- Associated HTTP XML characteristic type: “StandaloneMsg”

Node: /<x>/CPM/StandaloneMsg/MaxSize

Leaf node that represents the maximum authorised size of the content payload of a text or multimedia message without transfer encoding

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

Table 175: CPM MO sub tree addition parameters (MaxSize)

- Values: <content maximum size in bytes>
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “MaxSize”

Node: /<x>/CPM/MessageStore

Interior node where there are filled parameters related to RCS CPM Common Message Store

This node is not instantiated if the Service Provider does not provide the Common Message Store.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	node	Get

Table 176: CPM MO sub tree addition Message Store node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-cpm:6.0:MessageStore*
- Associated HTTP XML characteristic type: “MessageStore”

Node: /<x>/CPM/MessageStore/Url

Leaf node that represents the URL address of the Message Store Server

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	chr	Get

Table 177: CPM MO sub tree addition parameters (Url)

- Values: the URL for accessing the Message Store Server
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “Url”

Node: /<x>/CPM/MessageStore/AuthProt

Optional leaf node that can be used to force the Message Store Client to use one of the authentication methods defined in section 2.13.1.5.

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

Table 178: CPM MO sub tree addition parameters (AuthProt)

- Values: 0, 1, 2
  0- Indicates that the user name / password method must be used by the Message Store Client (default)
  2- Indicates that the authentication with the Common Message Store shall be based on the GBA security association.

NOTE: The value ‘1’ is not used for historic reasons

- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “AuthProt”

Node: /<x>/CPM/MessageStore/UserName

Optional leaf node that represents the User Identity information used by the Message Store Client to access the subscriber IMAP account on the Message Store Server

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	chr	No Get, No Copy

Table 179: CPM MO sub tree addition parameters (UserName)

- Values: <username assigned to the user for access to the Message Store Server>
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “UserName”

Node: /<x>/CPM/MessageStore/UserPwd

Optional leaf node that represents the user password associated to his/her User Name Identity information used by the Message Store Client to access the subscriber IMAP account on the Message Store Server

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	chr	No Get, No Copy

Table 180: CPM MO sub tree addition parameters (UserPwd)

- Values: <password assigned to the user for access to the Message Store Server>
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “UserPwd”

Node: /<x>/CPM/MessageStore/EventRpting

Optional leaf node that can be used to inform the Message Store Client whether to directly set flags in the Message Store or whether to indicate to the Messaging Server that it should set flags in the Message Store on behalf of the client. If not instantiated, the Message Store Client SHALL assume the same method as if value 0 had been specified.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	bool	Get

Table 181: CPM MO sub tree addition parameters (EventRpting)

- Values: 0, 1
  0- Indicates that the client shall set flags in IMAP as needed via an IMAP connection (default)
  1- Indicates that the client shall make use of the Event Reporting framework as described in section 4.1.4.8 when no IMAP connection exists so that the Messaging Server may set the flags in the Message Store on behalf of the client
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “EventRpting”

Node: /<x>/CPM/MessageStore/AuthArchive

Optional leaf node that can be used to enable the Message Store Client to archive messages.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	bool	Get

Table 182: CPM MO sub tree addition parameters (AuthArchive)

- Values: 0, 1
  0- Indicates that the archive service is disabled (default)
  1- Indicates that archive service is enabled and thus the client may archive messages.
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1
- Associated HTTP XML parameter ID: “AuthArchive”

Node: /<x>/CPM/MessageStore/SyncTimer

Leaf node that represents maximum time interval between two periodic client-triggered synchronisations towards the Message Store Server.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

Table 183: CPM MO sub tree addition parameters (SyncTimer)

- Values: <Timer value in seconds>
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “SyncTimer”

Node: /<x>/CPM/MessageStore/DataConnectionSyncTimer

Leaf node that represents minimum time interval between two data connection state related triggered synchronisations towards the Message Store Server.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

Table 184: CPM MO sub tree addition parameters (DataConnectionSyncTimer)

- Values: <Timer value in seconds>
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “DataConnectionSyncTimer”

Node: /<x>/CPM/MessageStore/SMSStore

Leaf node that describes whether the client is expected to store to the Message Store Server sent or received SMS.

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

Table 185: CPM MO sub tree addition parameters (SMSStore)

- Values: This parameter can have 3 possible values:
  0- The device shall not store any sent or received SMS to the Message Store Server
  1- The device shall store to the Message Store every sent and received SMS that cannot be correlated with the Common Message Store in the RCS Default folder
  2- The device shall store every sent and received SMS and shall not attempt to correlate with the Common Message Store in the RCS Default folder.
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “SMSStore”

Node: /<x>/CPM/MessageStore/MMSStore

Leaf node that describes whether the client is expected to store to the Message Store Server sent or received MMS.

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

Table 186: CPM MO sub tree addition parameters (MMSStore)

- Values: This parameter can have 3 possible values:
  0- The device shall not store any sent or received MMS to the Message Store Server
  1- The device shall store to the Message Store every sent and received MMS that cannot be correlated with the Common Message Store
  2- The device shall store every sent and received MMS and shall not attempt to correlate with the Common Message Store.
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “MMSStore”

Node: /<x>/CPM/Ext

An extension node for Service Provider specific parameters. Clients that are not aware of any extensions in this subtree (e.g. because they are not Service Provider specific) should not instantiate this tree.

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

Table 187: CPM MO sub tree addition Service Provider Extension Node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-cpm:6.0:EXT*
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML characteristic type: “Ext”

A.2.8. Capability discovery MO sub tree

This RCS specification includes the following additions as a new configuration sub tree, the capability discovery MO sub tree. Please note this sub tree is not included in any other specifications. So no other nodes from those specifications need to be added:

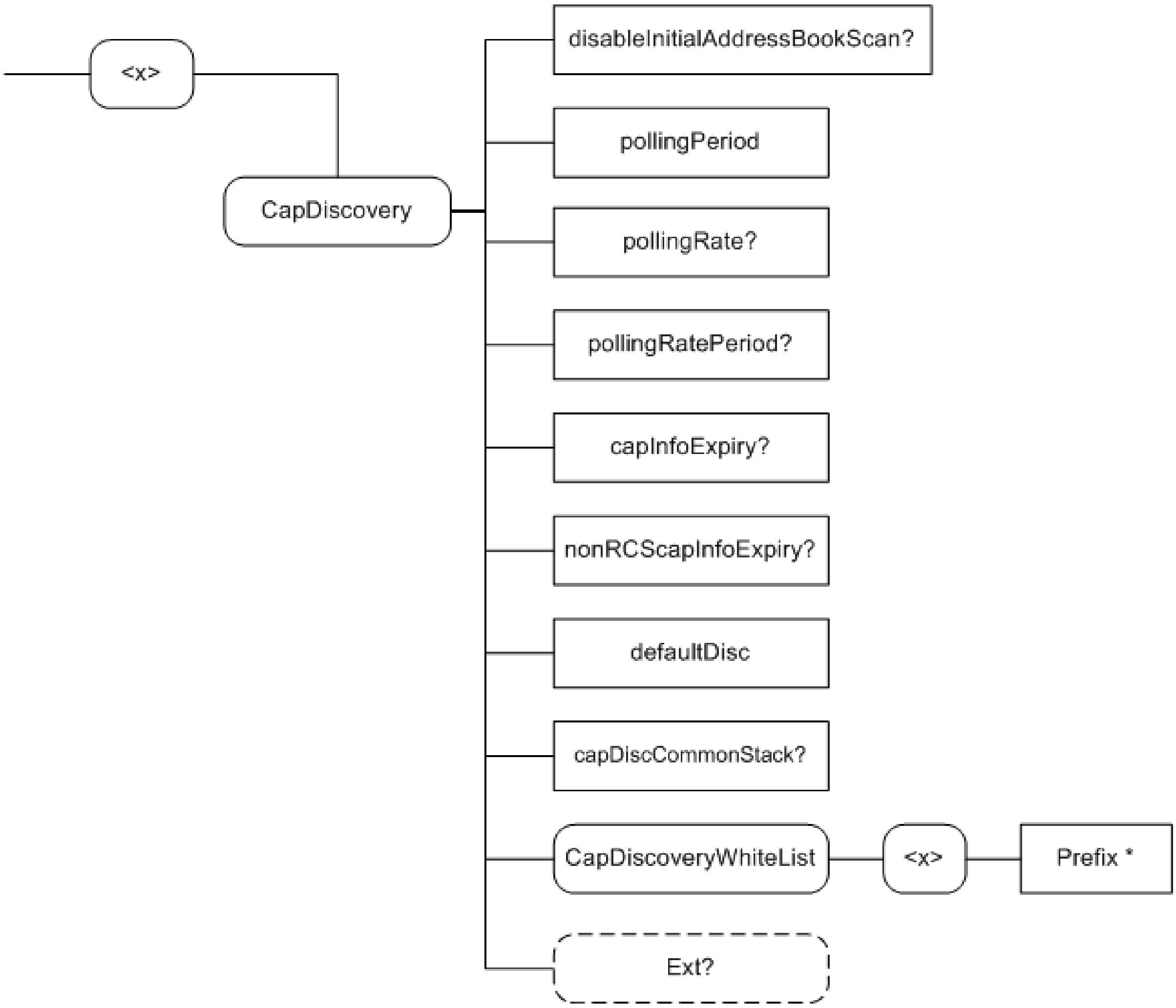

Figure 126: RCS additions, capability sub tree

The associated HTTP configuration XML structure is presented in the table below:

```xml
<characteristic type="CAPDISCOVERY">
        <parm name="disableInitialAddressBookScan" value="X"/>
        <parm name="pollingPeriod" value="X"/>
        <parm name="pollingRate" value="X"/>
        <parm name="pollingRatePeriod" value="X"/>
        <parm name="capInfoExpiry" value="X"/>
        <parm name="nonRCScapInfoExpiry" value="X"/>
        <parm name="defaultDisc" value="X"/>
        <parm name="capDiscCommonStack" value="X"/>
        <characteristic type="CapDiscoveryWhitelist">
                <characteristic type="CapDiscoveryAllowedPrefixes">
                        <parm name="Prefix1" value="X"/>
                        <parm name="Prefix2" value="X"/>
                        <parm name="Prefix3" value="X"/>
                        …
                </characteristic>
        </characteristic>
        <characteristic type="Ext"/>
</characteristic>
```

Table 188 : Capability sub tree associated HTTP configuration XML structure

Node: /<x>/CapDiscovery

Under this interior node the RCS parameters related to capability discovery are placed

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 189: Capability MO sub tree addition capability discovery node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-icapdis:6.0*
- Associated HTTP XML characteristic type: “CAPDISCOVERY”

Node: /<x>/CapDiscovery/disableInitialAddressBookScan

Leaf node that describes whether the device/client should when it is first started, perform a capability exchange for all contacts in the address book

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

Table 190: Capability MO sub tree addition parameters (disableInitialAddressBookScan)

- Values:
  0, The scan of the address book shall be done
  1, The scan of the address book shall not be done.
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “disableInitialAddressBookScan”

Node: /<x>/CapDiscovery/pollingPeriod

Leaf node that describes the timer in seconds between querying all the contacts in the address book to update the capabilities.

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

Table 191: Capability MO sub tree addition parameters (pollingPeriod)

- Values: The time in seconds. If it is set to 0, the periodic capability update (polling) is not performed and only the initial address book polling is performed.
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “pollingPeriod”

Node: /<x>/CapDiscovery/pollingRatePeriod

Leaf node that indicates, in seconds, the period duration for the calculation of the authorised number of capability query requests during this period

It shall be instantiated in case PollingPeriod is set to a value greater than zero.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

Table 192: Capability MO sub tree addition parameters (pollingRatePeriod)

- Values: The period in seconds (has to be greater than 0 when set).
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “pollingRatePeriod”

Node: /<x>/CapDiscovery/pollingRate

Leaf node that indicates the maximum capability query operations that are authorised globally for the User Agent during each period (period parameter defined in the previous pollingRatePeriod node).

It shall be instantiated in case PollingPeriod is set to a value greater than zero.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

Table 193: Capability MO sub tree addition parameters (pollingRate)

- Values: integer that represents the total amount of capability query operations per each period, independently of the number of contacts that have to be query (has to be greater than 0 when set).

- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “pollingRate”

Node: /<x>/CapDiscovery/capInfoExpiry

Leaf node that describes the validity of the capability information stored in the terminal in seconds

It shall be instantiated in case PollingPeriod is set to a value greater than zero.

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

Table 194: Capability MO sub tree addition parameters (capInfoExpiry)

- Values: The time in seconds.
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “capInfoExpiry”

Node: /<x>/CapDiscovery/nonRCScapInfoExpiry

Leaf node that describes how long a non RCS contact shall be prevented from being queried for its capabilities.

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

Table 195: Capability MO sub tree addition parameters (nonRCScapInfoExpiry)

- Values: The time in seconds.
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “nonRCScapInfoExpiry”

Node: /<x>/CapDiscovery/defaultDisc

Leaf node that describes the default capability and new user discovery mechanism used by the terminal (Presence or Options).

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

Table 196: Capability MO sub tree addition parameters (defaultDisc)

- Values:
  0, the default mechanism employed for capability discovery and new users will be OPTIONS.
  1, the default mechanism employed for capability discovery and new users will be

Presence
2, the mechanism employed for capability discovery will be disabled.

- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “defaultDisc”

Node: /<x>/CapDiscovery/capDiscCommonStack

Leaf node that describes the interworking approach for the capability discovery. Please note this is only instantiated when the defaultDisc parameter is set to 1.

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

Table 197: Capability MO sub tree addition parameters (capDiscCommonStack)

- Values:
  0, the fallback to SIP OPTIONS mechanism remains disabled.
  1, the fallback to SIP OPTIONS mechanism remains enabled.
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “capDiscCommonStack”

Node: /<x>/CapDiscovery/CapDiscoveryWhiteList

A Placeholder interior node for the Capability Discovery white list configuration

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 198: Capability MO sub tree addition Capability Discovery White List node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-icapdis:6.0:capdiswhitelist*
- Associated HTTP XML characteristic type: “CapDiscoveryWhiteList”

Node: /<x>/CapDiscovery/CapDiscoveryWhiteList/<x>

A Placeholder interior node where to place 1 or more Prefix leaf nodes

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 199: Capability MO sub tree addition CapDiscoveryAllowedPrefixes node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-icapdis:6.0:capdiswhitelist:prefixes*
- Associated HTTP XML characteristic type: “CapDiscoveryAllowedPrefixes”

Node: /<x>/CapDiscovery/CapDiscoveryWhiteList/<x>/Prefix

Leaf node that represent a prefix configured by the Service Provider

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrMore	chr	No Get, No Copy

Table 200: Capability MO sub tree addition parameters (Prefix)

- Values: <a Service Provider defined prefix>.
  The prefix is used to identify phone numbers considered for the capability discovery mechanism. For the purpose of matching a phone number with the prefix the client shall remove visual separators and white space from the input phone number string. The phone numbers are identified by matching with the prefix value starting from the left. Its length can be one or more digits and it can start with the "+" character.
  Examples: +446, +4479, 00446, 004479, 06, 079
  In addition the client shall support matching of phone numbers based on regular expression. To indicate that the Prefix parameter value contains a regular expression it shall start with the "!" character. The subsequent string shall be interpreted by the client to match the phone number string using POSIX extended regular expression (see [POSIX]).
  Example: !^(\+44|0044|0)(6|79)\d*
- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: “Prefix<X>” where <X> is a positive integer value

Node: /<x>/CapDiscovery/Ext

An extension node for Service Provider specific parameters. Clients that are not aware of any extensions in this subtree (e.g. because they are not Service Provider specific) should not instantiate this tree.

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

Table 201: Capability MO sub tree addition Service Provider Extension Node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-icapdis:6.0:Ext*
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML characteristic type: “Ext”

A.2.9. APN Configuration MO sub tree

This RCS specification includes the following additions as a new configuration sub tree, the roaming MO sub tree. Please note this sub tree is not included in any other specifications. So no other nodes from those specifications need to be added:

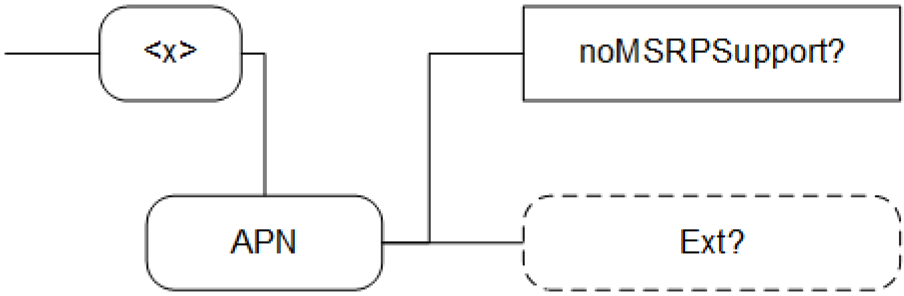

Figure 127: RCS additions, roaming sub tree

The associated HTTP configuration XML structure is presented in the table below:

```
<characteristic type="APN">
        <parm name="noMSRPSupport" value="X"/>
        <characteristic type="Ext"/>
</characteristic>
```

Table 202 : APN sub tree associated HTTP configuration XML structure

Node: /<x>/APN

Under this interior node the RCS parameters related to roaming are placed.

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 203: APN MO sub tree addition node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-apn:6.0*
- Associated HTTP XML characteristic type: "APN"

Node: /<x>/APN/noMSRPSupport

Leaf node that lists roaming partners the home operators knows do not support MSRP. When MSRP is not supported when roaming, the client shall use the HOS APN for RCS if obtained through client configuration; otherwise, shall use the Internet APN.

If not instantiated, the device shall behave as if it was an empty list.

In case the device cannot support the IMS APN, it shall ignore this parameter.

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	Chr	Get, Replace

Table 204: APN MO sub tree addition parameters (noMSRPSupport)

- Values: <a comma separated list of names identifying a roaming partners>
  This parameter is used to list roaming partners that do not support MSRP. The parameter shall be composed of a comma-separated list containing each roaming partner's MCC (Mobile Country Code) followed by ".", then followed by the MNC (Mobile Network Code) whereby <MNC> and <MCC> shall be replaced by the respective values of the Service Provider's network in decimal format and with a 2-digit MNC padded out to 3 digits by inserting a 0 at the beginning (as defined in

[PRD-IR.67]).
Example of a list with two roaming partners: mnc<mnc>.mcc<mcc>, mnc<mnc>.mcc<mcc>

- 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 (see section 2.4) using the new parameter.
- Associated HTTP XML parameter ID: "noMSRPSupport"

Node: /<x>/APN/Ext

An extension node for Service Provider specific parameters. Clients that are not aware of any extensions in this subtree (e.g. because they are not Service Provider specific) should not instantiate this tree.

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

Table 205: APN MO sub tree addition Service Provider Extension Node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-apn:6.0:Ext*
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML characteristic type: "Ext"

A.2.10. Other RCS Configuration MO sub tree

This RCS specification includes the following additions as a new configuration sub tree, containing the remaining RCS configuration parameters. Please note this sub tree is not included in any other specifications. So no other nodes from those specifications need to be added:

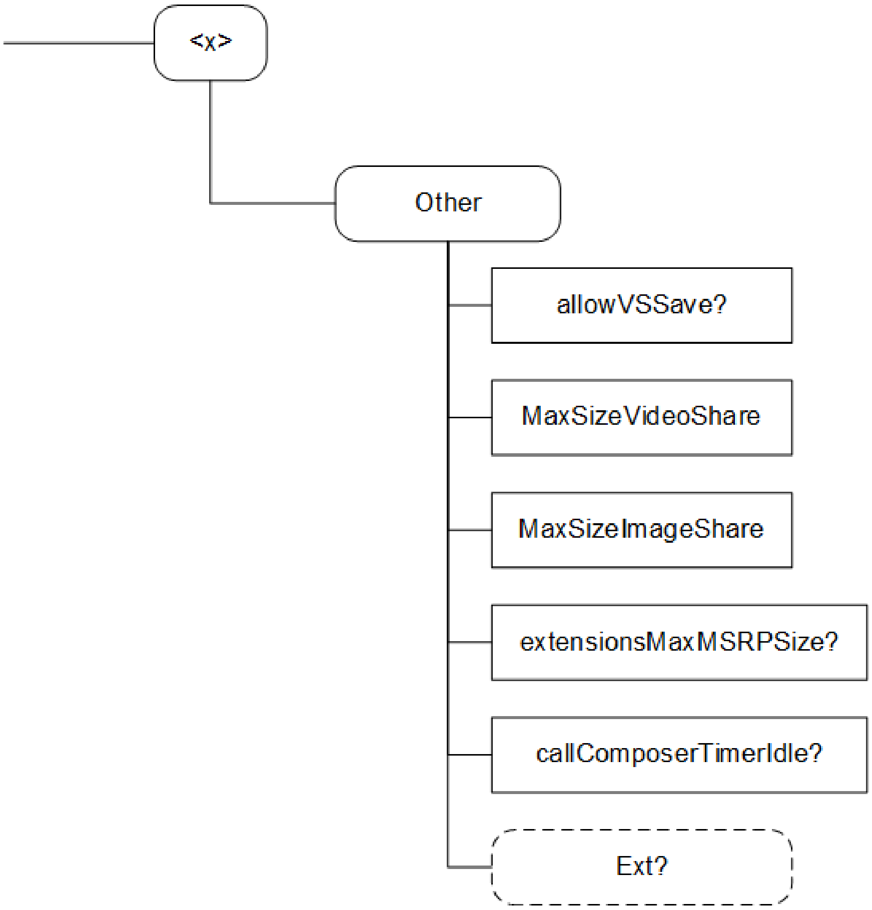

Figure 128: RCS additions, other sub tree

The associated HTTP configuration XML structure is presented in the table below:

```
<characteristic type="OTHER">
        <parm name="allowVSSave" value="X"/>
        <parm name="MaxSizeImageShare" value="X"/>
        <parm name="MaxTimeVideoShare value="X"/>
        <parm name="extensionsMaxMSRPSize" value="X"/>
        <parm name="callComposerTimerIdle" value="X"/>
        <characteristic type="Ext"/>
</characteristic>
```

Table 206 : Other sub tree associated HTTP configuration XML structure

Node: /<x>/Other

Under this interior node the RCS parameters which do not fit in the other categories are placed.

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

Table 207: Other MO sub tree addition node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-other:6.0*
- Associated HTTP XML characteristic type: “OTHER”

Node: /<x>/Other/allowVSSave

Leaf node that determines whether or not the SDP attribute and value described in section 3.6.4.1.1 is included in the Video and Image Share invitations

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

Table 208: Other MO sub tree addition parameters (allowVSSave)

- Values: -1, 0, 1
  -1- Inclusion of the attribute and value is up to the user's preference
  0- The attribute is never included (default if not provided)
  1- The attribute is always included
- Post-reconfiguration actions: Any change in the value shall be taken into account in the SDP provided during the setup of any new Video and Image Share sessions.
- Associated HTTP XML parameter ID: "allowVSSave"

Node: /<x>/MaxSizeImageShare

Leaf node that represents the maximum authorised size of the content that can be sent in an Image Share session

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

Table 209: IMS MO sub tree addition parameters (MaxSizeImageShare)

- Values: <content maximum size in bytes>. Value equals to 0 means no limitation.
- Post-reconfiguration actions: Any change in the value shall be taken into account in the setup of any new Image Share sessions.
- Associated HTTP XML characteristic type: "MaxSizeImageShare"

Node: /<x>/MaxTimeVideoShare

Leaf node that represents the maximum authorised duration time for a Video Share session

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

Table 210: IMS MO sub tree addition parameters (MaxTimeVideoShare)

- Values: <Timer value in seconds>. Value equals to 0 means no limitation.
- Post-reconfiguration actions: Any change in the value shall be taken into account in the any new Video Share sessions (i.e. they shall not affect an ongoing Video Share).
- Associated HTTP XML characteristic type: "MaxTimeVideoShare"

Node: /<x>/Other/extensionsMaxMSRPSize

Leaf node that represent the maximum authorised size of an Extension to Extension message exchanged via MSRP.

It is required to be instantiated if a service provider enables Extensions by configuring a value for the extensionsIMSRouting parameter.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	bool	Get

Table 211: Other MO sub tree addition parameters (extensionsMaxMSRPSize)

- Values: <content maximum size in bytes>
- Post-reconfiguration actions: Any change in the value shall be taken into account in the any new Extension to Extension sessions (i.e. such changes shall not affect an ongoing session).
- Associated HTTP XML characteristic type: “extensionsMaxMSRPSize”

Node: /<x>/Other/callComposerTimerIdle

As per section 2.1.2.1 of [PRD-RCC.20]

Node: /<x>/Other/Ext

An extension node for Service Provider specific parameters. Clients that are not aware of any extensions in this subtree (e.g. because they are not Service Provider specific) should not instantiate this tree.

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

Table 212: Other MO sub tree addition Service Provider Extension Node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-other:6.0:Ext*
- Post-reconfiguration actions: The client should be reset and should perform the complete first-time registration procedure following a reconfiguration (e.g. OMA-DM/HTTP) as described in section 2.3.1.1.
- Associated HTTP XML characteristic type: “Ext”

A.2.11. VV-Mail RCS Configuration MO sub tree

This RCS specification includes the following additions as a new configuration sub tree, the voice mail MO sub tree. Please note this sub tree is not included in any other specification. So no other nodes from those specifications need to be added:

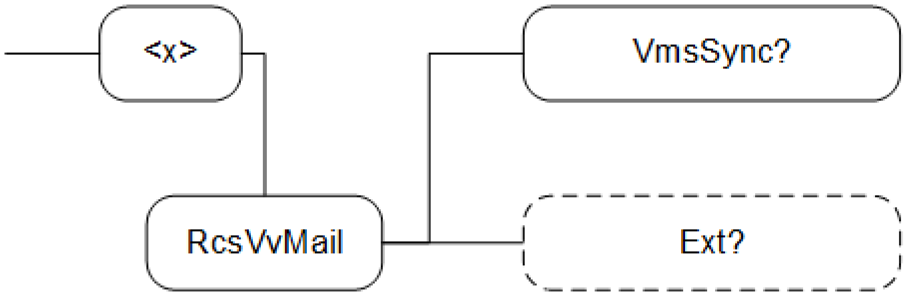

Figure 129: RCS additions, VV-Mail sub tree

The associated HTTP configuration XML structure is presented in the table below:

```
<characteristic type="RcsVvMail">
        <parm name="VmsSync" value="X"/>
        <characteristic type="Ext"/>
</characteristic>
```

Table 213 : Voicemail sub tree associated HTTP configuration XML structure

Node: /<x>/RcsVvMail

Under this interior node the RCS parameters related to voicemail are placed.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	node	Get

Table 214: Voicemail MO sub tree addition node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-vvmail:6.0*
- Associated HTTP XML characteristic type: “RcsVvMail”

Node: /<x>/RcsVvMail/VmsSync

Leaf node that represent the authorisation and synchronization of the RCS VV-Mail service for secondary devices.

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	bool	Get

Table 215: Voicemail MO sub tree addition parameter (VmsSync)

- Values: 0, 1
  0- The value indicates that the voicemail service is disabled and voice mail sync is disabled (default)
  1- The value indicates that the voicemail service and voicemail sync is enabled
- Post-reconfiguration actions: As the client remains unregistered during configuration, there are no additional actions apart from re-registering using the new parameter if the parameter value has changed.
- Associated HTTP XML parameter ID: “VmsSync”

Node: /<x>/RcsVvMail/Ext

An extension node for Service Provider specific parameters. Clients that are not aware of any extensions in this subtree (e.g. because they are not Service Provider specific) should not instantiate this tree.

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

Table 216: Voicemail MO sub tree addition Service Provider Extension Node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-vvmail:6.0:Ext*

- Associated HTTP XML characteristic type: “Ext”

A.2.12. Service Provider Extensions MO sub tree

This RCS specification includes the following additions as a new and optional configuration sub tree, the Service Provider extensions MO sub tree. This tree should not instantiate by clients that are not aware of any extensions in this tree. Please note this sub tree is not included in any other specifications. So no other nodes from those specifications need to be added:

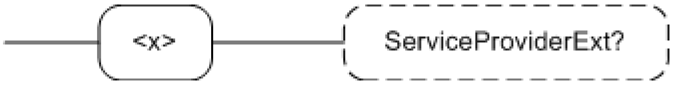

Figure 130: RCS additions, Service Provider Extensions sub tree

The associated HTTP configuration XML structure is presented in the table below:

```
<characteristic type=”SERVICEPROVIDEREXT”/>
```

Table 217 : Service Provider Extensions sub tree associated HTTP configuration XML structure

Node: /<X>/ServiceProviderExt

Under this interior node the RCS parameters related to Service Provider specific extensions are placed

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

Table 218: Service Provider Extensions MO sub tree addition node

- Values: N/A
- Type property of the node is: *urn:gsma:mo:rcs-sp:6.0*
- Associated HTTP XML characteristic type: “SERVICEPROVIDEREXT”

A.3. HTTP specific configuration and behaviour

A.3.1. HTTP configuration XML structure

In addition to the parameters and characteristics type correspondences presented in the previous section, it is necessary to define the following mandatory configuration XML elements[41]:

```
<?xml version="1.0"?>
<wap-provisioningdoc version="1.1">
        <characteristic type="VERS">
                <parm name="version" value="1"/>
                <parm name="validity" value="1728000"/>
        </characteristic>
        <characteristic type="TOKEN">
                <parm name="token" value="X"/>
        </characteristic>
        <characteristic type="MSG">                    -- This section is OPTIONAL
                <parm name="title" value="Example"/>
                <parm name="message" value="Hello world"/>
                <parm name="Accept_btn" value="X"/>
                <parm name="Reject_btn" value="X"/>
        </characteristic>
        <characteristic type="APPLICATION">
                <parm name="AppID" value="ap2001"/>
                <parm name="Name" value="IMS Settings"/>
                <parm name="AppRef" value="IMS-Settings"/>
                        …        -- see section [PRD-RCC.15]
        </characteristic>
        <characteristic type="APPLICATION">
                <parm name="AppID" value="ap2002"/>
                <parm name="Name" value="RCS settings"/>
                <parm name="AppRef" value="RCSe-Settings"/>
                <characteristic type="IMS">
                        <parm name="To-AppRef" value="IMS-Settings"/>
                </characteristic>
                <characteristic type="SERVICES">
                        …        -- See section A.2.1
                </characteristic>
                <characteristic type="PRESENCE">
                        …        -- See section A.2.3
                </characteristic>
                <characteristic type="XDMS">
                        …        -- See section A.2.4
                </characteristic>
                <characteristic type="SUPL">
                        …        -- See section A.2.5
                </characteristic>
                <characteristic type="IM">
                        …        -- See section A.2.6
                </characteristic>
                <characteristic type="CPM">
                        …        -- See section A.2.7
```

[41] Please note the AppID's used in the example are provided for reference only as they have not been reserved.

```
            </characteristic>
            <characteristic type=”CAPDISCOVERY”>
                …          -- See section A.2.8
            </characteristic>
            <characteristic type=”APN”>
                …          -- See section A.2.9
            </characteristic>
            <characteristic type=”OTHER”>
                …          -- See section A.2.10
            </characteristic>
            <characteristic type=”RcsVvMail”>
                …          -- See section A.2.11
            </characteristic>
            <characteristic type=”SERVICEPROVIDEREXT”>
                …          -- See section A.2.12
            </characteristic>
        </characteristic>
</wap-provisioningdoc>
```

Table 219: Complete RCS HTTP configuration XML structure

A.4. Autoconfiguration XML sample

```
<?xml version="1.0"?>
<wap-provisioningdoc version="1.1">
        <characteristic type="VERS">
              <parm name=”version” value=”1”/>
              <parm name=”validity” value=”1728000”/>
        </characteristic>
        <characteristic type="TOKEN">
              <parm name=”token” value=”X”/>
        </characteristic>
        <characteristic type="MSG">                    -- This section is OPTIONAL
              <parm name=”title” value=”Example”/>
              <parm name=”message” value=”Hello world”/>
              <parm name=”Accept_btn” value=”X”/>
              <parm name=”Reject_btn” value=”X”/>
        </characteristic>                              -- This section is OPTIONAL
        <characteristic type="APPLICATION">
              <parm name=”AppID” value=”ap2001”/>
              <parm name=”Name” value=”IMS Settings”/>
              <parm name=”AppRef” value=”IMS-Settings”/>
              …                                       -- see [PRD-RCC.15]
        </characteristic>
        <characteristic type="APPLICATION">
              <parm name=”AppID” value=”ap2002”/>
              <parm name=”Name” value=”RCS settings”/>
              <parm name=”AppRef” value=”RCSe-Settings”/>
              <characteristic type=”IMS”>
                    <parm name=”To-AppRef” value=”IMS-Settings”/>
              </characteristic>
              <characteristic type=”SERVICES”>
                    <parm name=”presencePrfl” value=”X”/>
                    <parm name=”ChatAuth” value=”X”/>
                    <parm name=”GroupChatAuth” value=”X”/>
                    <parm name=”ftAuth” value=”X”/>
                    <parm name=”standaloneMsgAuth” value=”X”/>
                    <parm name=”geolocPullAuth” value=”X”/>
                    <parm name=”geolocPushAuth” value=”X”/>
                    <parm name=”vsAuth” value=”X”/>
                    <parm name=”isAuth” value=”X”/>
                    <parm name=”rcsIPVoiceCallAuth” value=”X”/>
                    <parm name=”rcsIPVideoCallAuth” value=”X”/>
                    <parm name=”IR94VideoAUTH” value=”X”/>
                    <parm name=”allowRCSExtensions” value=”X”/>
                    <parm name=”composerAuth” value=”X”/>
                    <parm name=”sharedMapAuth” value=”X”/>
                    <parm name=”sharedSketchAuth” value=”X”/>
                    <parm name=”postCallAuth” value=”X”/>
                    <parm name=”IR51VoiceAuth” value=”X”/>
                    <parm name=”IR51VideoAuth” value=”X”/>
                    <parm name=”VvmAuth” value=”X”/>

-- Continues in the next table --
```

Table 220: Complete RCS autoconfiguration XML structure (1/4)

-- Follows from previous table –

```
                <characteristic type=”Ext”>
                        <characteristic type=”DataOff”>
                            <parm name=”rcsMessagingDataOff” value=”X”/>
                            <parm name=”fileTransferDataOff” value=”X”/>
                            <parm name=”smsoIPDataOff” value=”X”/>
                            <parm name=”mmsDataOff” value=”X”/>
                            <parm name=”contentShareDataOff” value=”X”/>
                            <parm name=”preAndPostCallDataOff” value=”X”/>
                            <parm name=”volteDataOff” value=”X”/>
                            <parm name=”IPVideoCallDataOff” value=”X”/>
                            <parm name=”provisioningDataOff” value=”X”/>
                            <parm name=”extensionsDataOff” value=”X”/>
                            <parm name=”syncDataOff” value=”X”/>
                            <characteristic type=”Ext”/>
                        </characteristic>
                    </characteristic>
            </characteristic>
            <characteristic type=”PRESENCE”>
                <parm name=”NoteMaxSize” value=”X”/>
                <characteristic type=”VIPCONTACTS”>
                        <parm name=”NonVipPollPeriodSetting” value=”X”/>
                        <parm name=”NonVipMaxPollPerPeriod” value=”X”/>
                </characteristic>
                <parm name=”NickNameLength” value=”X”/>
                <parm name=”LocInfoMaxValidTime” value=”X”/>
                <characteristic type=”Ext”/>
                <parm name=”client-obj-datalimit” value=”X”/>
                <parm name=”source-throttlepublish” value=”X”/>
                <parm name=”max-number-ofsubscriptions-inpresence-list” value=”X”/>
                <parm name=”RLS-URI” value=”X”/>
            </characteristic>
            <characteristic type=”XDMS”>
                <parm name=”RevokeTimer” value=”X”/>
                <parm name=”enablePNBManagement” value=”X”/>
                <characteristic type=”Ext”/>
                <parm name=”XCAPRootURI” value=”X”/>
                <parm name=”XCAPAuthenticationUserName” value=”X”/>
                <parm name=”XCAPAuthenticationSecret” value=”X”/>
                <parm name=”XCAPAuthenticationType” value=”X”/>
            </characteristic>
            <characteristic type=”SUPL”>
                <parm name=”TextMaxLength” value=”X”/>
                <parm name=”LocInfoMaxValidTime” value=”X”/>
                <parm name=”geolocPullBlockTimer” value=”X”/>
                <characteristic type=”Ext”/>
            </characteristic>
            <characteristic type=”IM”>
                <parm name=”imMsgTech” value=”X”/>
                <parm name=”imCapAlwaysON” value=”X”/>
                <parm name=”imCapNonRCS” value=”X”/>
```

-- Continues in the next table --

Table 221: Complete RCS autoconfiguration XML structure (2/4)

-- Follows from previous table –

```
            <characteristic type="GroupChatNonRCSWhitelist">
                  <parm name="imCapNonRCSGroupChat" value="X"/>
                  <parm name="imCapNonRCSLimitGroupChat" value="X"/>
                  <characteristic type="GroupChatAllowedPrefixes">
                        <parm name=" Prefix1" value="X"/>
                        <parm name=" Prefix2" value="X"/>
                        <parm name=" Prefix3" value="X"/>
                        …
                  </characteristic>
            </characteristic>
             <parm name="AutAccept" value="X"/>
            <parm name="imSessionStart" value="X"/>
            <parm name="AutAcceptGroupChat" value="X"/>
            <parm name="TimerIdle" value="X"/>
            <parm name="MaxSize" value="X"/>
            <parm name="ChatRevokeTimer" value="X"/>
            <parm name="ftWarnSize" value="X"/>
            <parm name="MaxSizeFileTr" value="X"/>
            <parm name="MaxSizeFileTrIncoming" value="X"/>
            <parm name="ftCapAlwaysON" value="X"/>
            <parm name="ftAutAccept" value="X"/>
            <parm name="ftHTTPCSURI" value="X"/>
            <parm name="ftHTTPCSUser" value="X"/>
            <parm name="ftHTTPCSPwd" value="X"/>
            <parm name="ftDefaultMech" value="X"/>
            <parm name="ftHTTPFallback" value="X"/>
            <characteristic type="Ext"/>
            <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>
      <characteristic type="CPM">
            <characteristic type="StandaloneMsg">
                  <parm name="MaxSizeStandalone" value="X"/>
            </characteristic>
            <characteristic type="MessageStore">
                  <parm name="Url" value="X"/>
                  <parm name="AuthProt" value="X"/>
                  <parm name="UserName" value="X"/>
                  <parm name="UserPwd" value="X"/>
                  <parm name="EventRpting" value="X"/>
                  <parm name="AuthArchive" value="X"/>
                  <parm name="SyncTimer" value="X"/>
                  <parm name="DataConnectionSyncTimer" value="X"/>
                  <parm name="SMSStore" value="X"/>
                  <parm name="MMSStore" value="X"/>
            </characteristic>
            <characteristic type="Ext"/>
      </characteristic>
```

-- Continues in the next table -

Table 222: Complete RCS autoconfiguration XML structure (3/4)

```
-- Follows from previous table –

            <characteristic type="CAPDISCOVERY">
                <parm name="disableInitialAddressBookScan" value="X"/>
                <parm name="pollingPeriod" value="X"/>
                <parm name="pollingRate" value="X"/>
                <parm name="pollingRatePeriod" value="X"/>
                <parm name="capInfoExpiry" value="X"/>
                <parm name="nonRCScapInfoExpiry" value="X"/>
                <parm name="defaultDisc" value="X"/>
                <parm name="capDiscCommonStack" value="X"/>
                <characteristic type="CapDiscoveryWhitelist">
                    <characteristic type="CapDiscoveryAllowedPrefixes">
                        <parm name="Prefix1" value="X"/>
                        <parm name="Prefix2" value="X"/>
                        <parm name="Prefix3" value="X"/>
                        …
                    </characteristic>
                </characteristic>
                <characteristic type="Ext"/>
            </characteristic>
            <characteristic type="APN">
                <parm name="noMSRPSupport" value="X"/>
                <characteristic type="EXT"/>
            </characteristic>
            <characteristic type="OTHER">
                <parm name="allowVSSave" value="X"/>
                <parm name="MaxSizeImageShare" value="0"/>
                <parm name="MaxTimeVideoShare" value="0"/>
                <parm name="extensionsMaxMSRPSize" value="X"/>
                <parm name="callComposerTimerIdle" value="X"/>
                <characteristic type="Ext"/>
            </characteristic>
            <characteristic type="RcsVvMail">
                <parm name="VmsSync" value="X"/>
                <characteristic type="Ext"/>
            </characteristic>
            <characteristic type="SERVICEPROVIDEREXT"/>
        </characteristic>
</wap-provisioningdoc>
```

Table 223: Complete RCS autoconfiguration XML structure (4/4)

Annex B: Additional diagrams

B.1. Chat and store and forward diagrams without Auto-Accept

B.1.1. Chat without store and forward

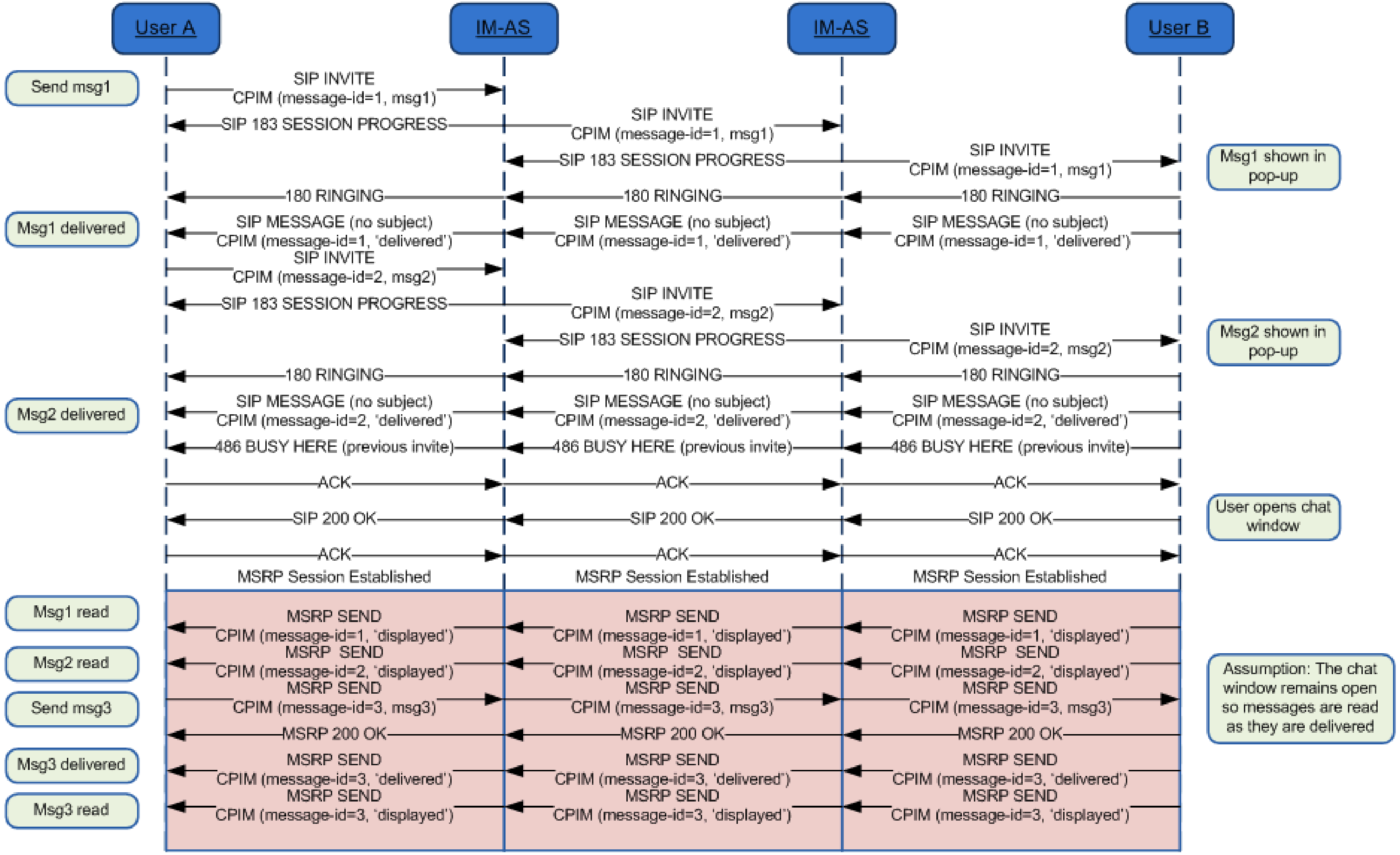

Figure 131: Chat flow without store and forward *

*: Check NOTE 1 and 15 in section B.1.19

B.1.2. Store and forward: Receiver offline

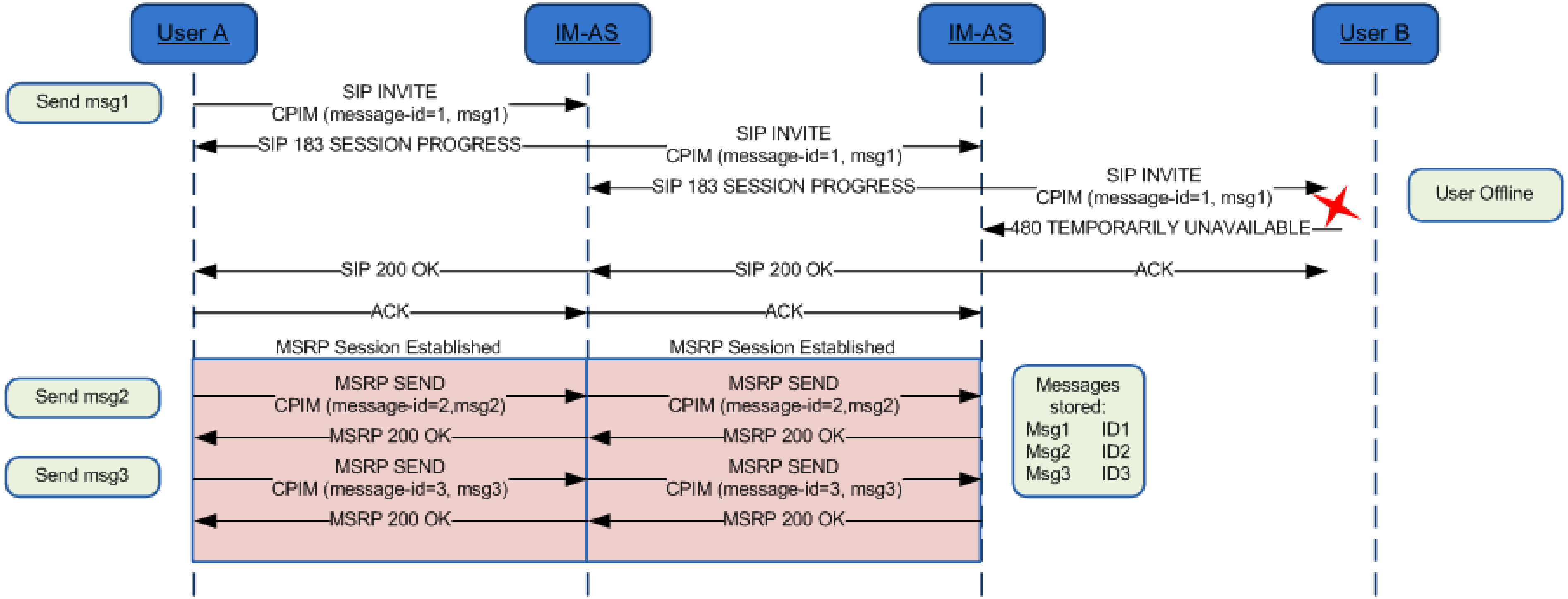

Figure 132: Store and forward: Receiver offline*

*: Check NOTE 1, 6 and 15 in section B.1.19

B.1.3. Store and forward: Message deferred delivery with sender still on an active Chat session

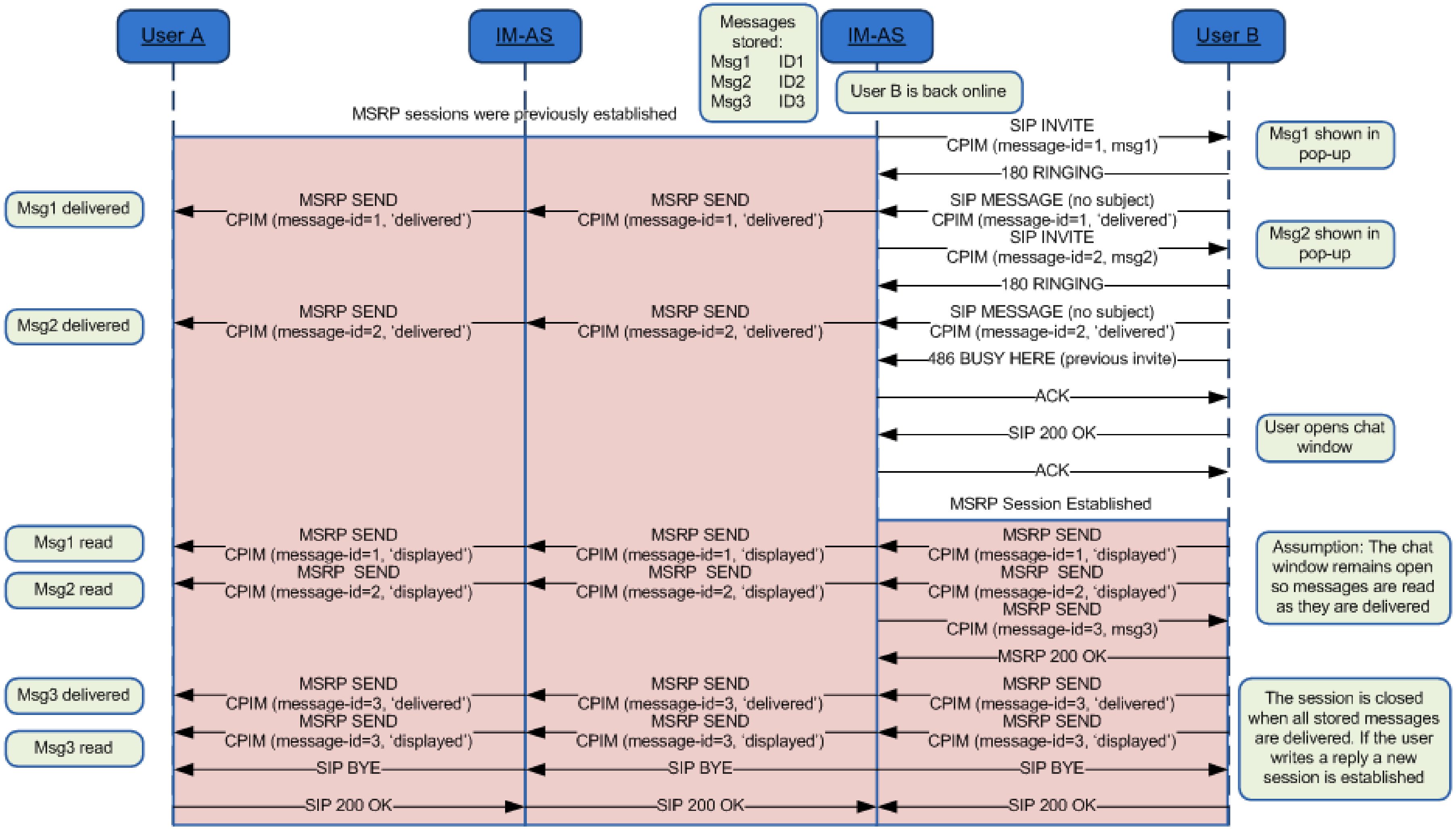

Figure 133: Store and forward: Message(s) deferred delivery with a sender still on an MSRP session*

*: Check NOTES 1, 2, 4, 7, 11 and 15 in section B.1.19

B.1.4. Store and forward: Message deferred delivery with sender online

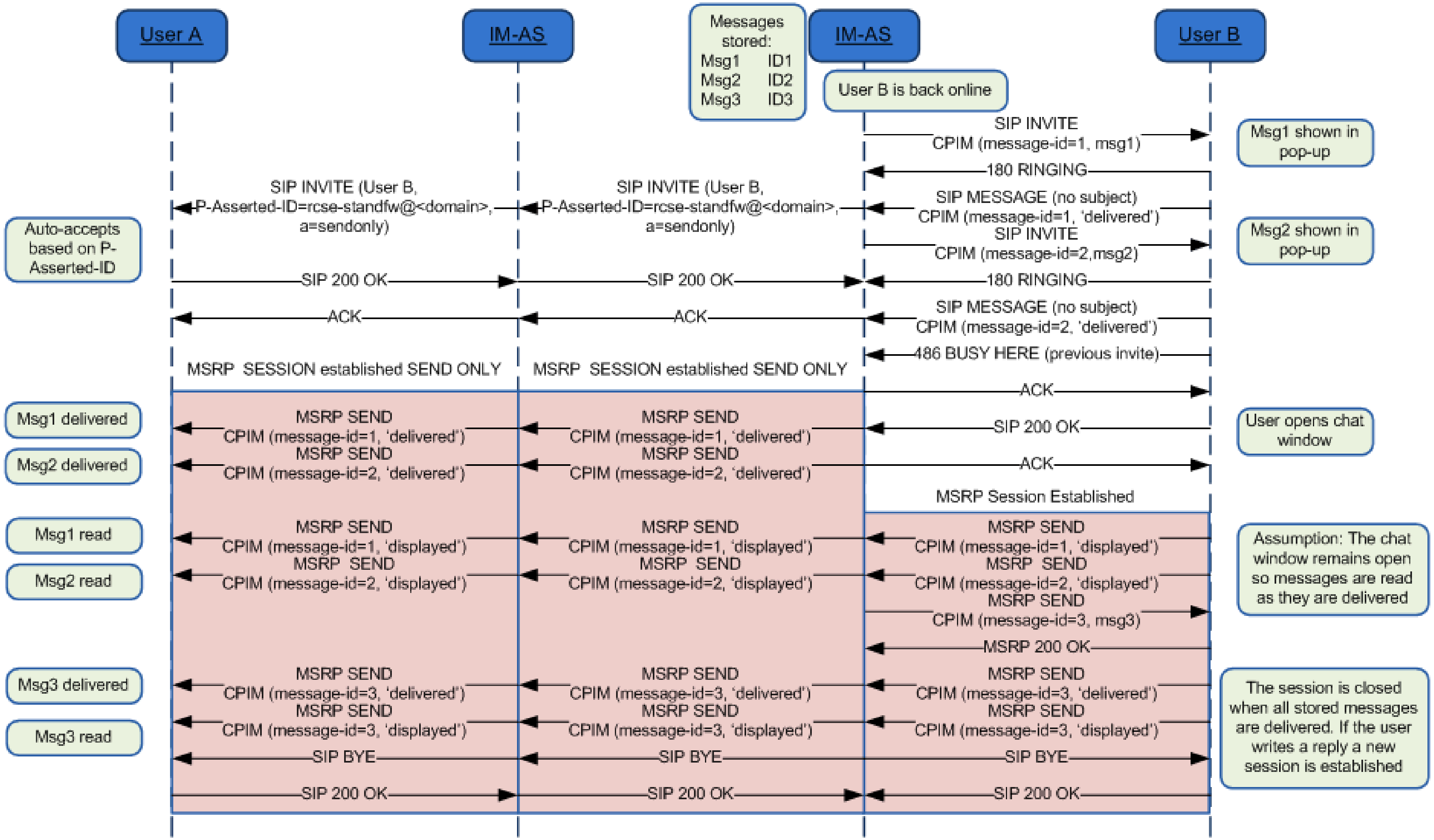

Figure 134: Store and forward: Message deferred delivery with sender online *

*: Check NOTES 1, 3, 4, 5, 7, 11, 14 and 15 in section B.1.19

B.1.5. Store and forward: Message deferred delivery with sender offline (delivery notifications)

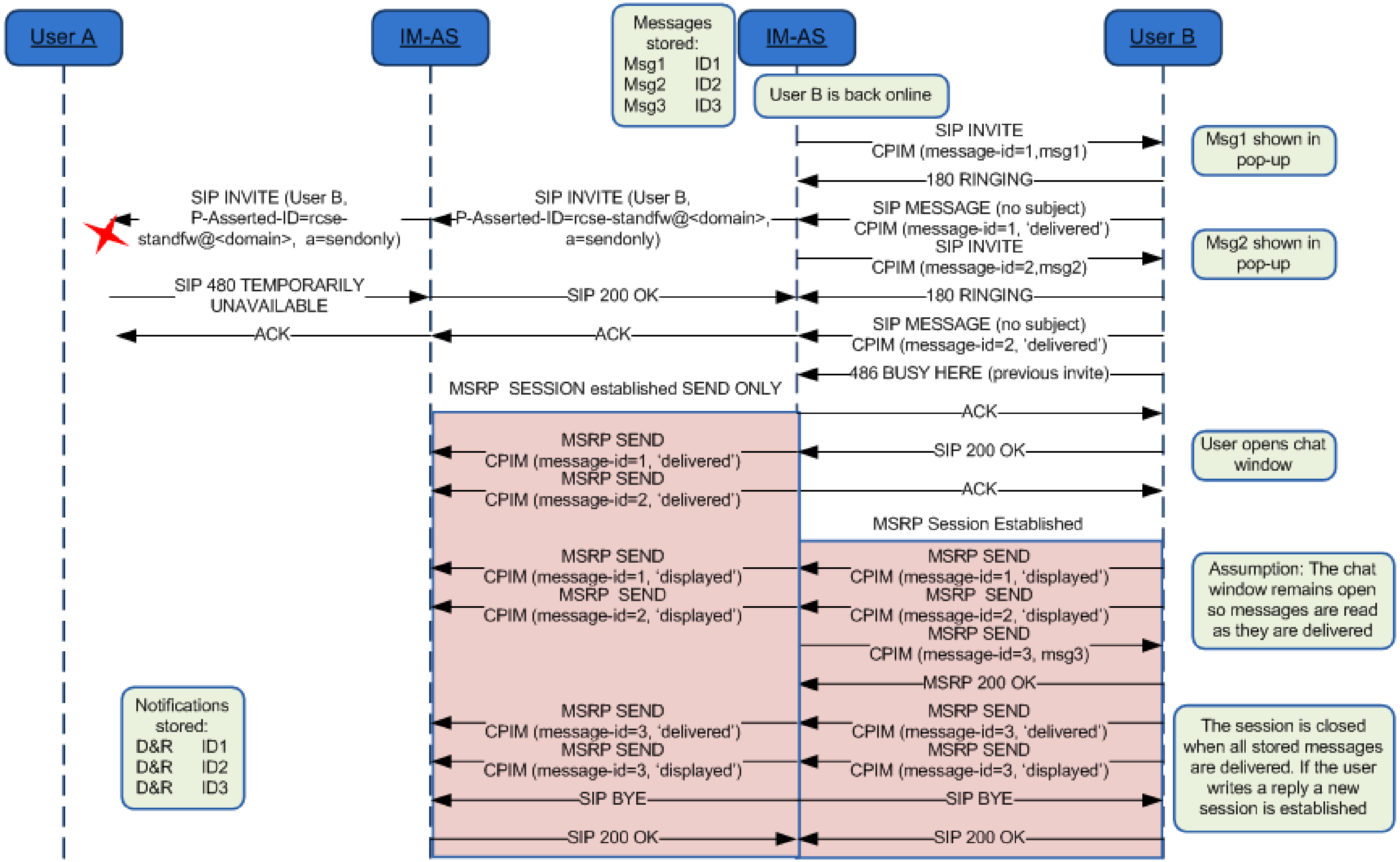

Figure 135: Store and forward: Message(s) deferred delivery with a sender offline (delivery notifications)*

*: Check NOTE 1, 5, 7, 11, 14 and 15 in section B.1.19

B.1.6. Store and forward: Notifications deferred delivery

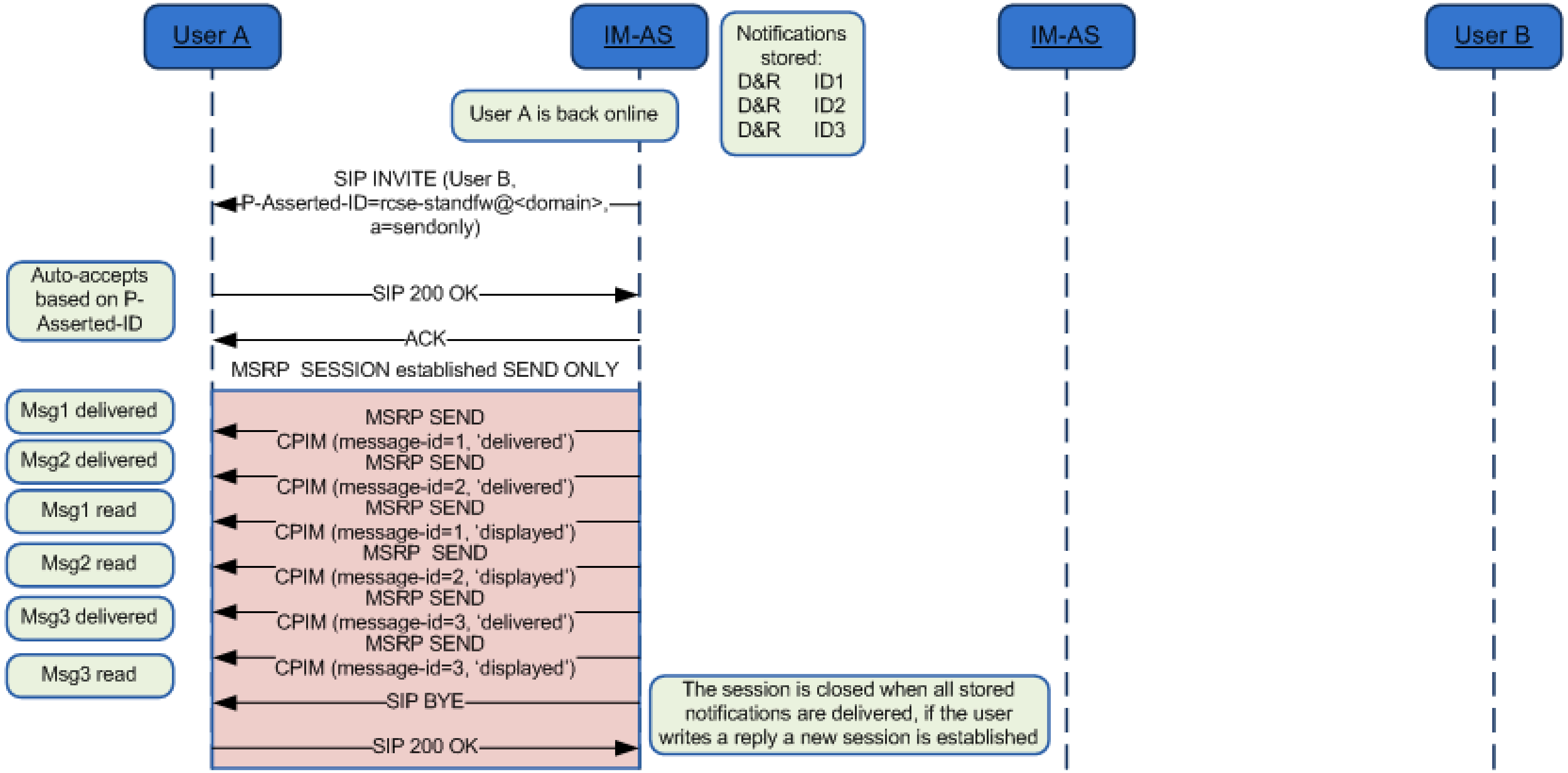

Figure 136: Store and forward: Notification(s) deferred delivery*

*: Check NOTES 1, 4, 5, 11, 14 and 15 in section B.1.19

B.1.7. Delivery of displayed notifications in an unanswered chat (without store and forward)

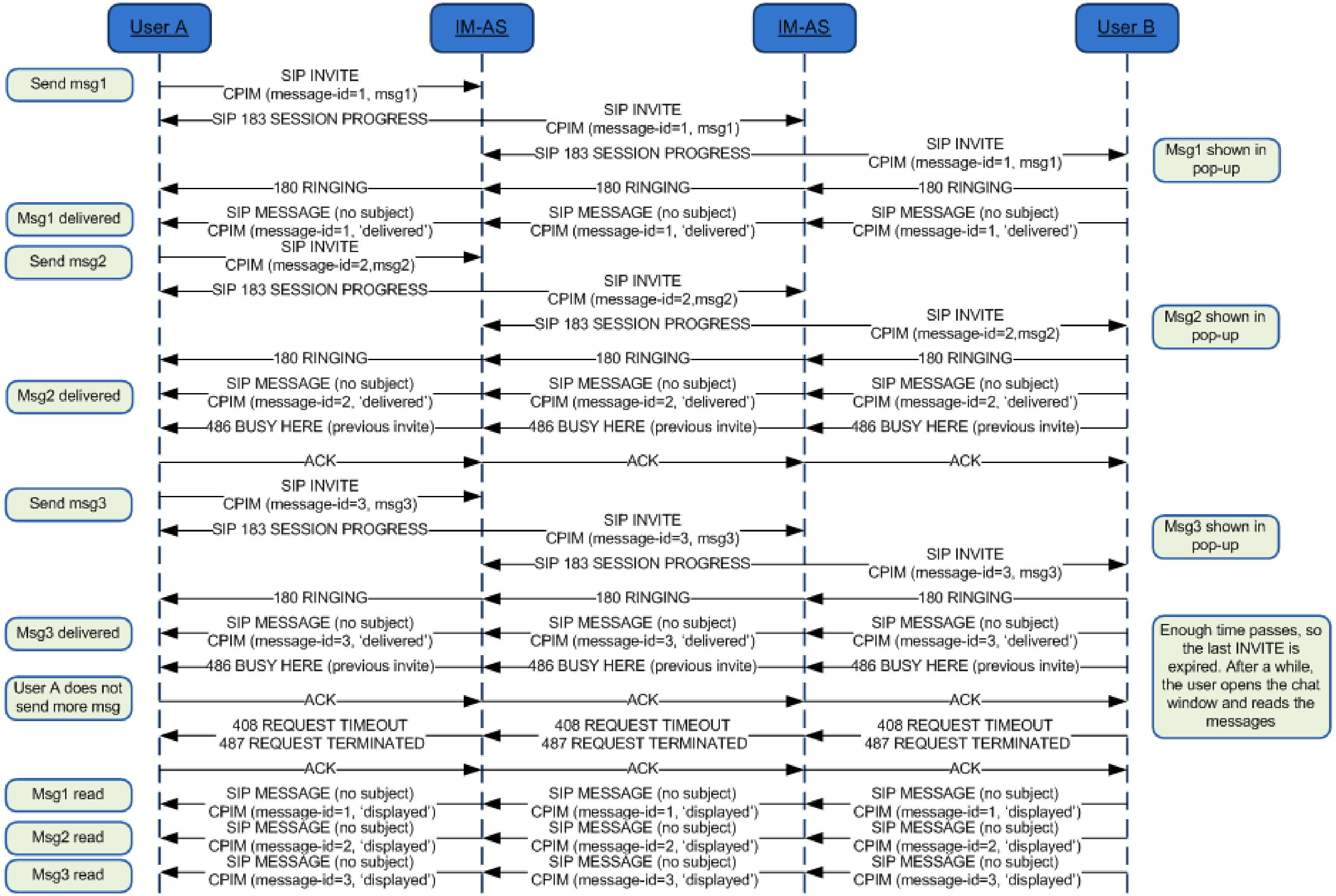

Figure 137: Delivery of displayed notifications in an unanswered chat (without store and forward)*

*: Check NOTE 1, 10 and 15 in section B.1.19

B.1.8. Store and forward: Handling errors in the receiver's side

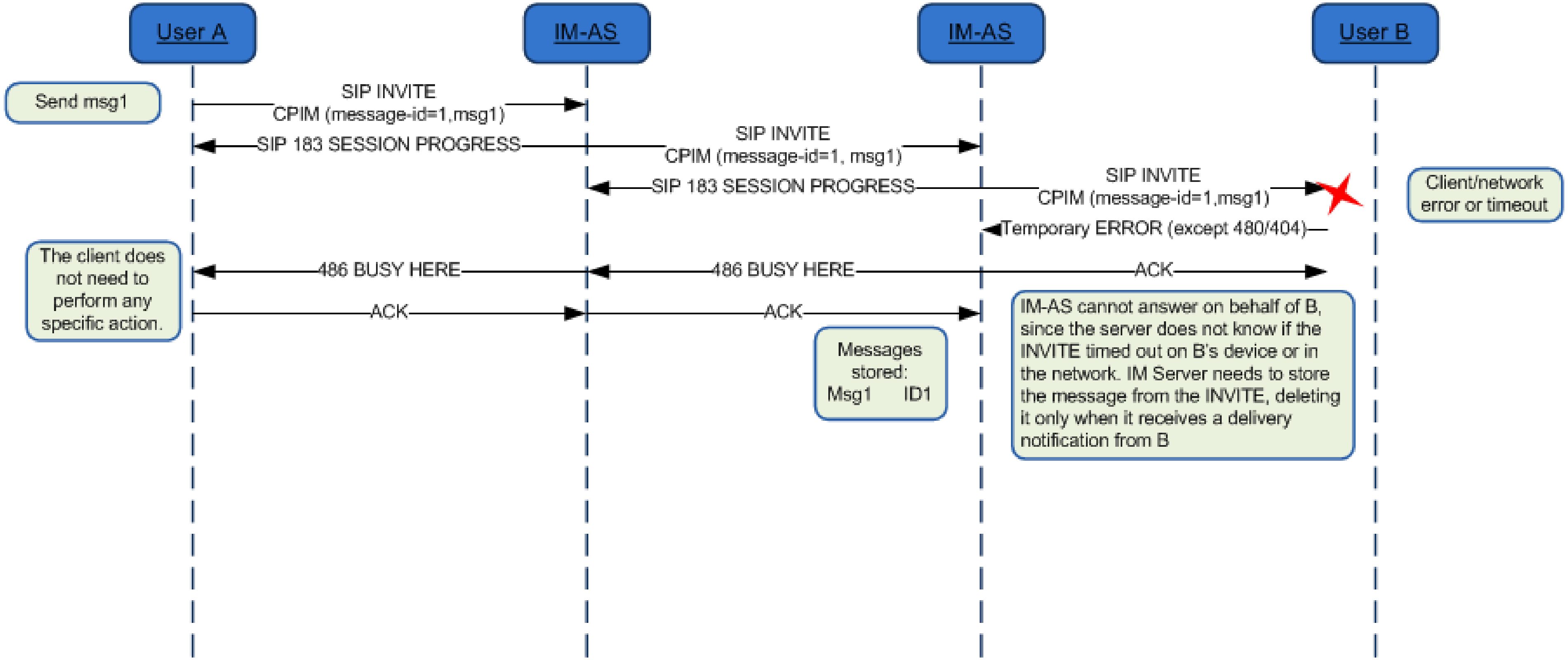

Figure 138: Store and forward: Handling errors in the receiver's side*

*: Check NOTE 15 in section B.1.19

NOTE: The error messages that are mapped to 486 Busy Here are listed in Table 31.

Also on the path between the IM-ASs (Instant Messaging Application Server i.e. the Messaging Server) similar errors could occur. In that case if the originating Messaging Server supports Store and Forward, it will behave in the same way and store the message.

B.1.9. Race conditions: Simultaneous INVITEs

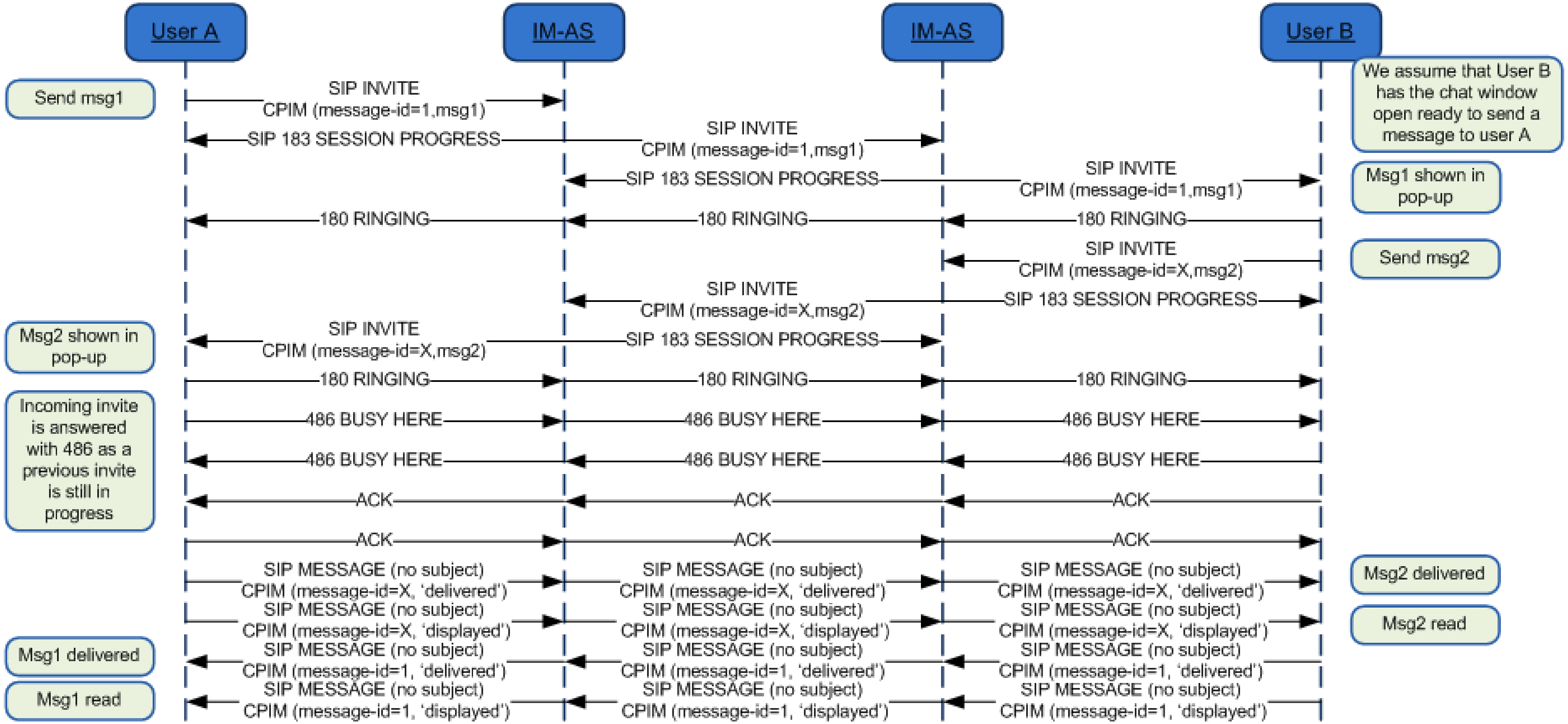

Figure 139: Store and forward race conditions: Simultaneous INVITEs*

*: Check NOTE 1 and 15 in section B.1.19

B.1.10. Race conditions: New INVITE after a session is accepted

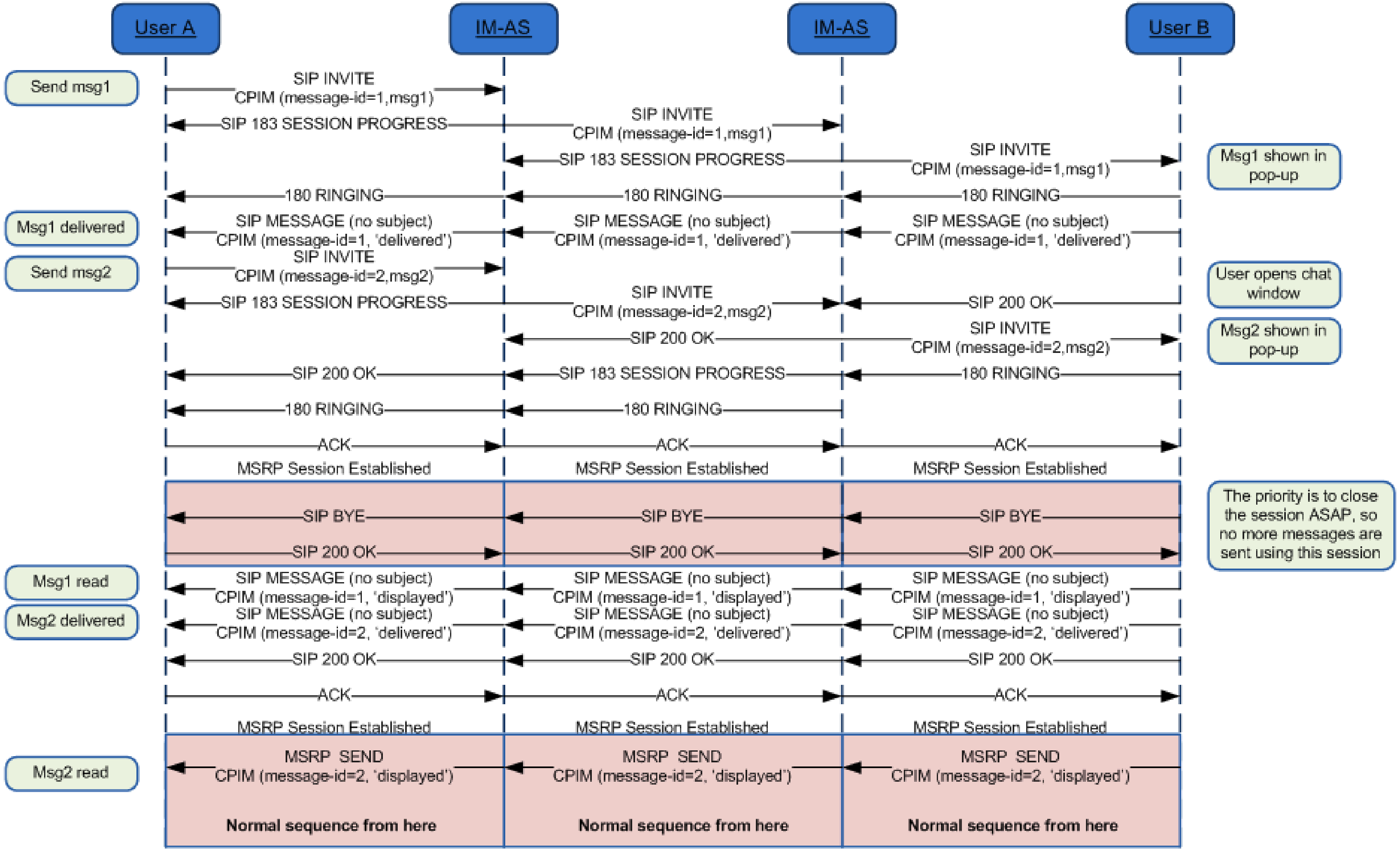

Figure 140: Store and forward race conditions: New INVITE after a session is accepted*

*: Check NOTE 1 and 15 in section B.1.19

B.1.11. Store and forward: Message(s) displayed notifications via SIP MESSAGE with sender offline

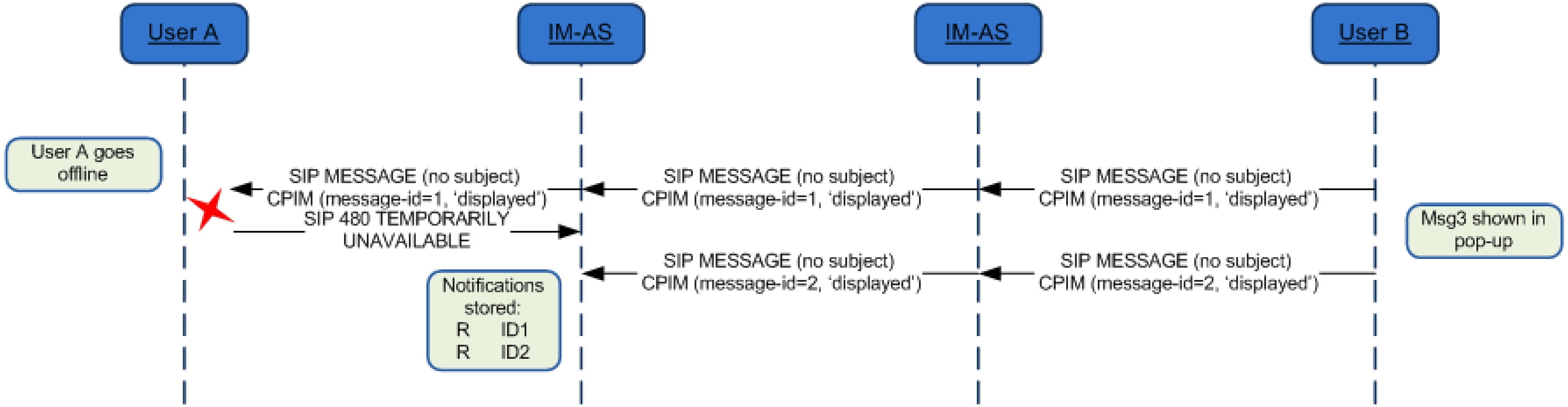

Figure 141: Store and forward: Message(s) displayed notifications via SIP MESSAGE with sender offline*

*: Check NOTES 1, 8, 9, 10 and 15 in section B.1.19

B.1.12. Interworking to SMS/MMS with automatic accept at the IWF

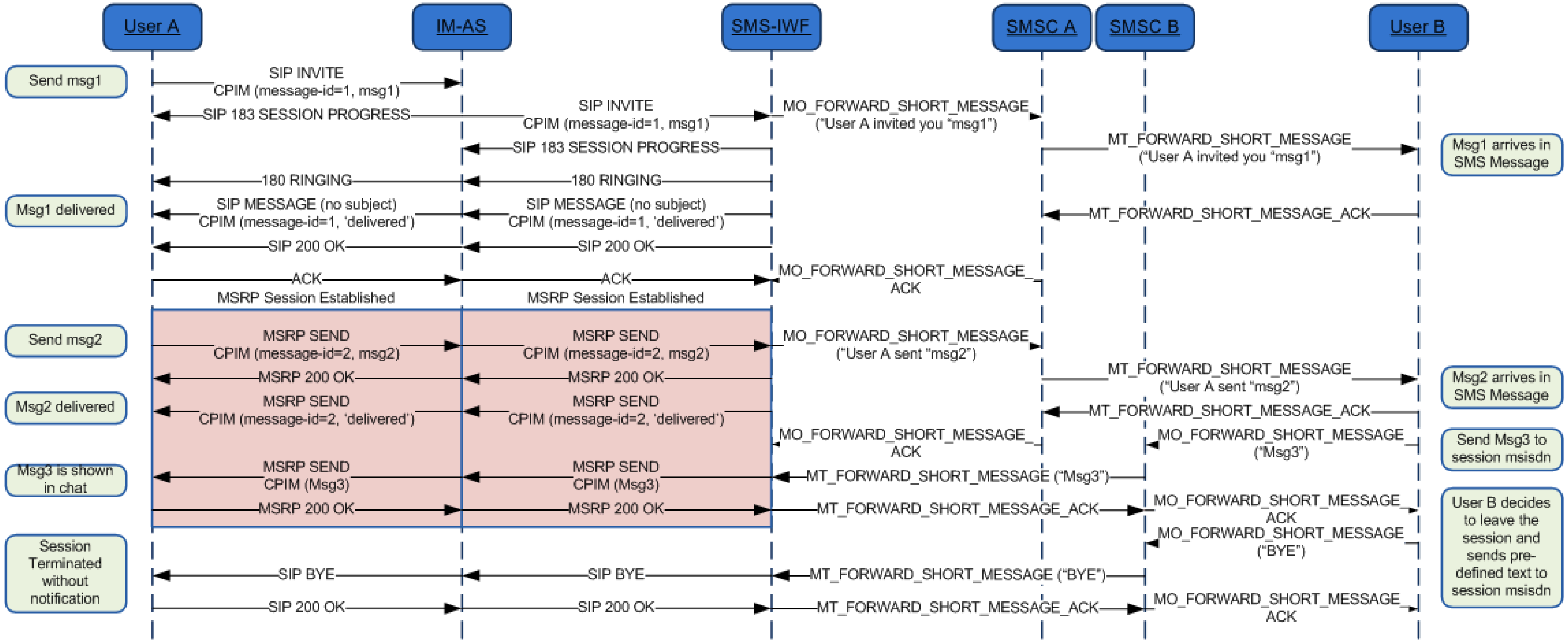

Figure 142: Interworking: Automatic acceptance on behalf of the SMS/MMS user*

*: Check NOTES 1, 12, 15 and 16 in section B.1.19

B.1.13. Interworking to SMS/MMS with manual accept

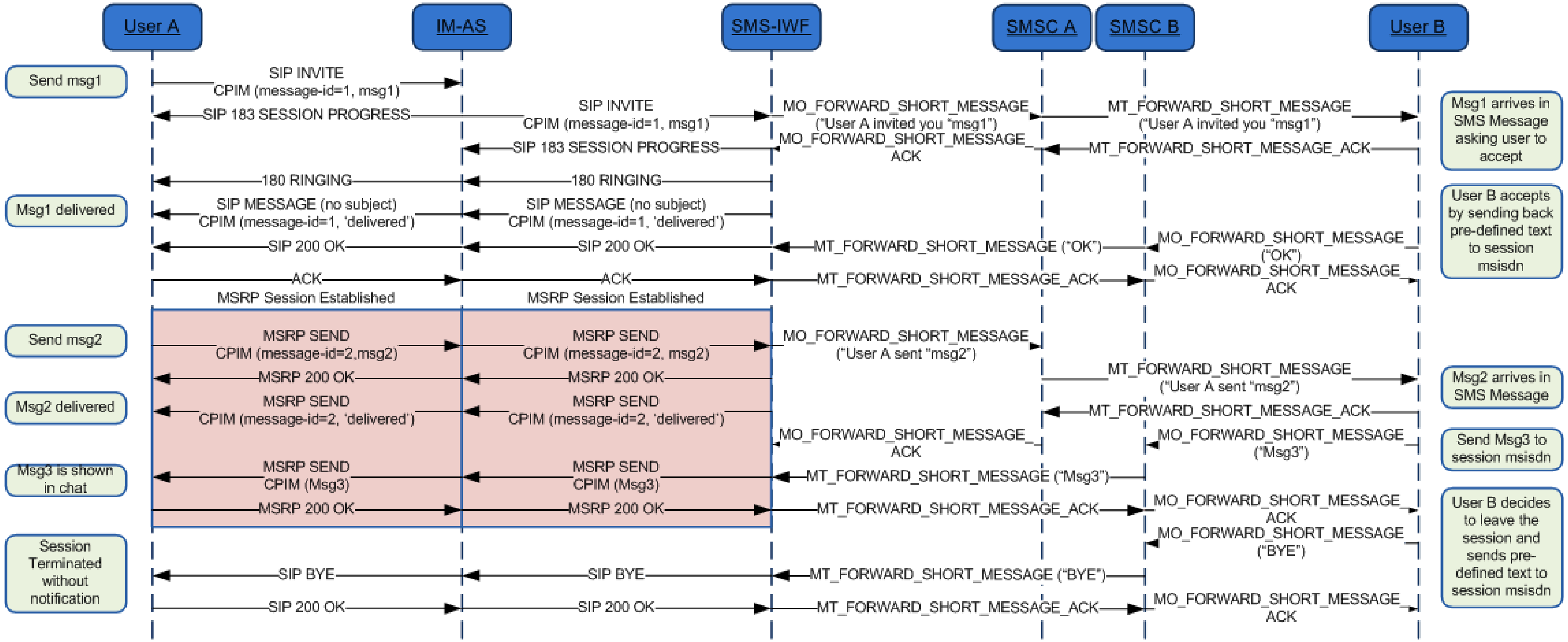

Figure 143: Interworking: manual acceptance by the SMS/MMS user*

*: Check NOTES 1, 12, 13, 15 and 16 in section B.1.19

B.1.14. Message Revoke: Successful Request

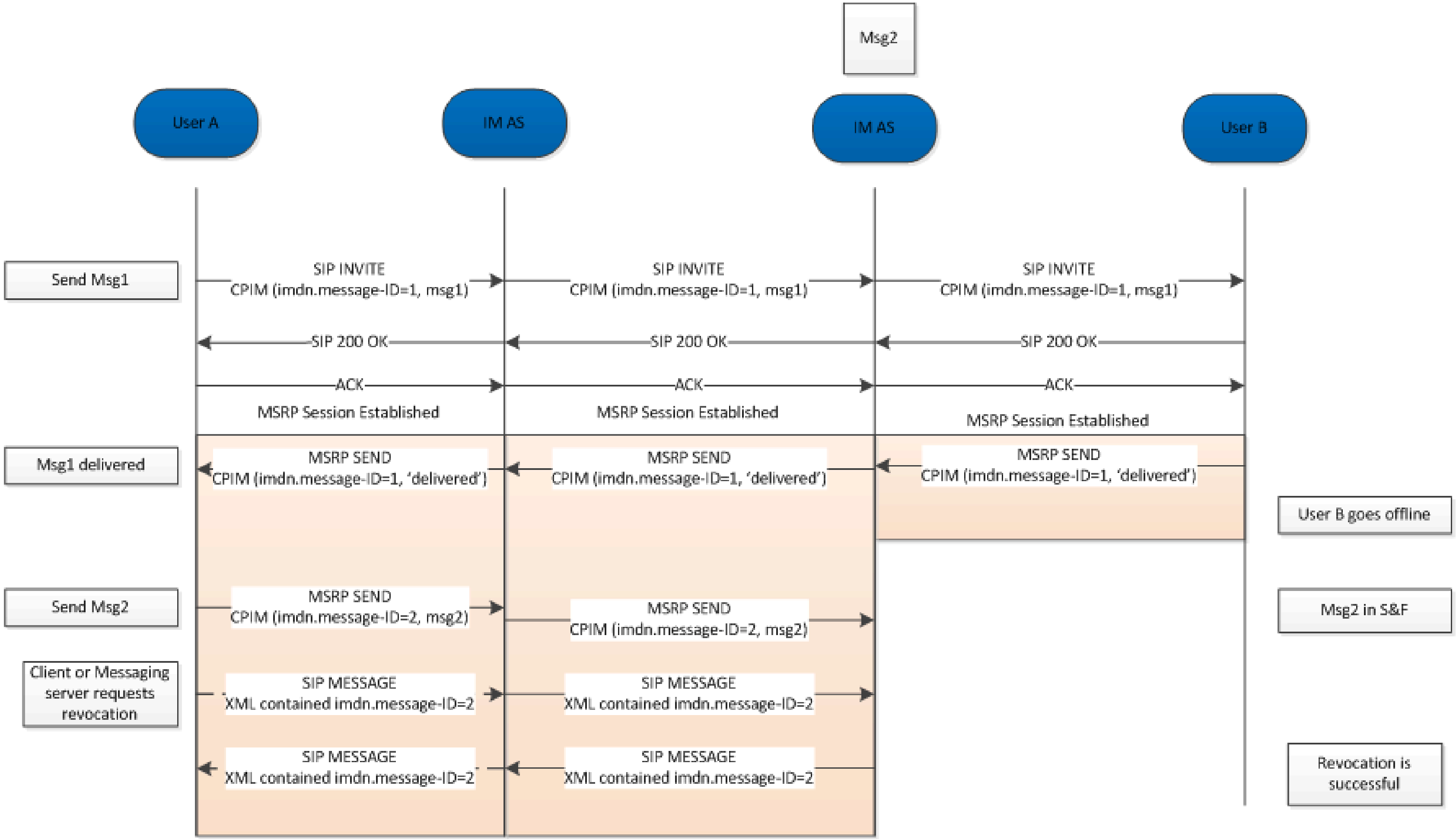

Figure 144: Message Revoke, Successful request*

*: Check NOTES 1, 6, 7 and 15 in section B.1.19

B.1.15. Message Revoke: Failed Request

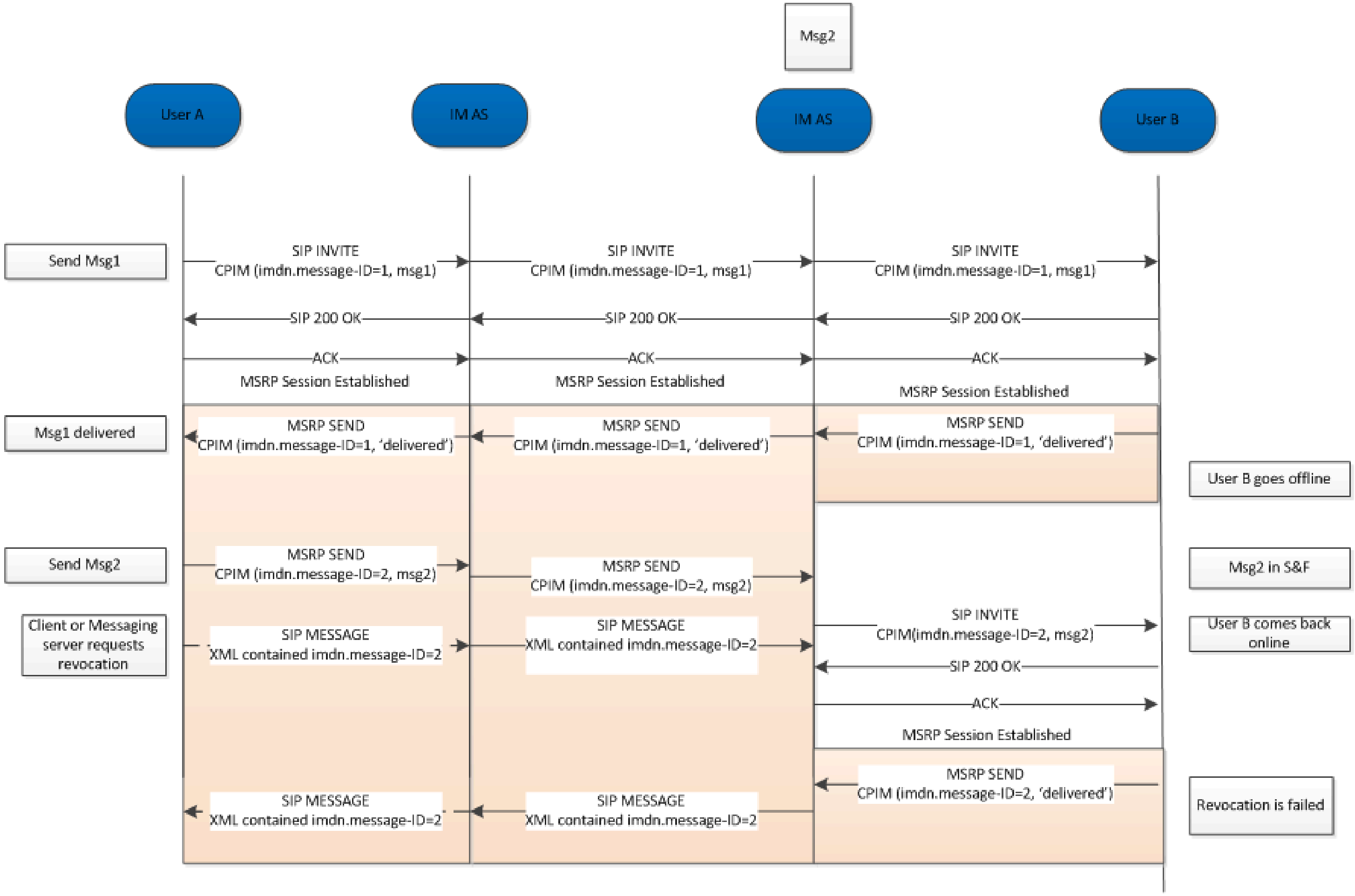

Figure 145: Message Revoke, Failed request*

*: Check NOTES 1, 6, 7 and 15 in section B.1.19

B.1.16. Re-joining a Group Chat that timed out due to inactivity

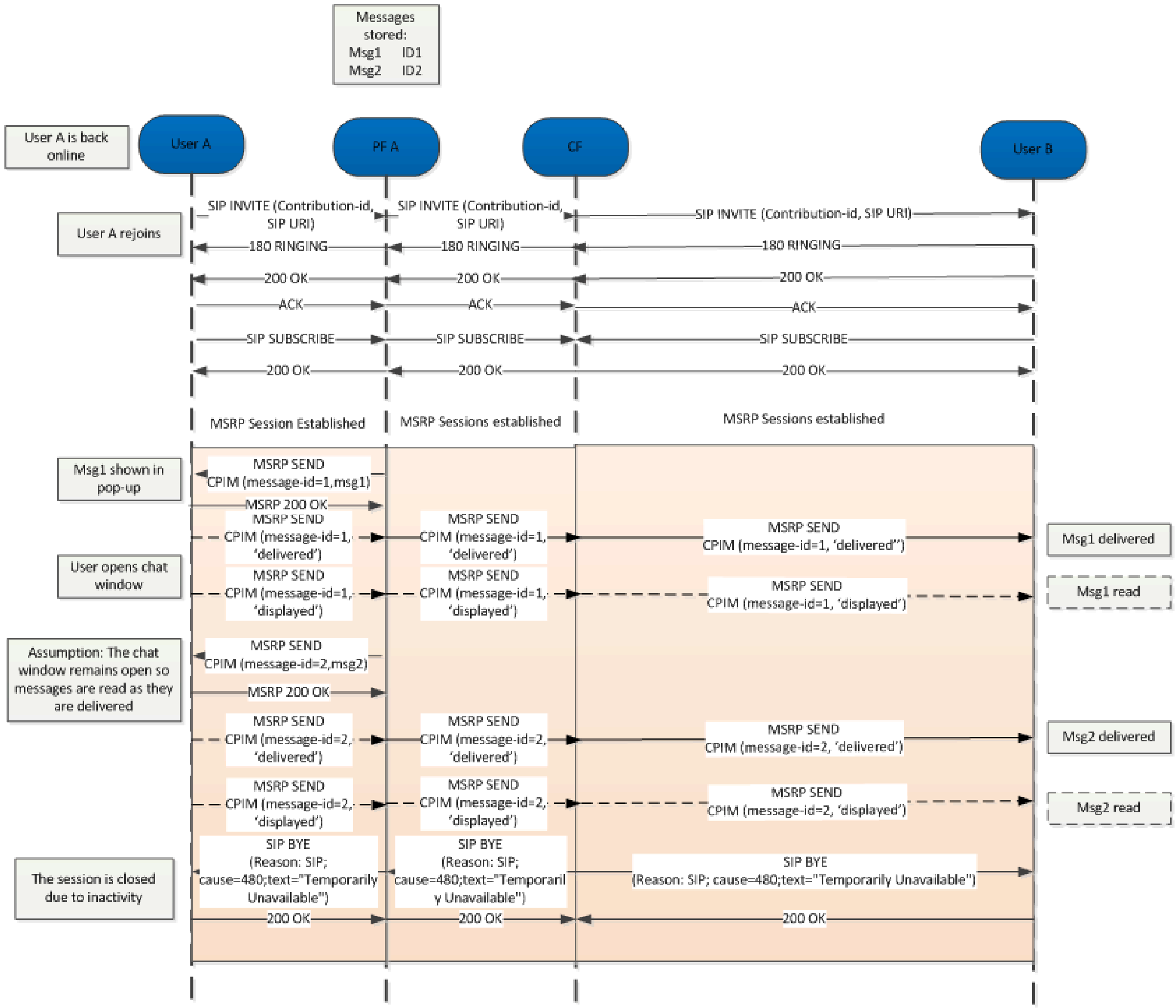

Figure 146: Rejoining a Group Chat that timed out due to inactivity *

*: Check NOTES 1, 15 and 17 in section B.1.19

B.1.17. Deliver Group Chat Messages while Chat is idle

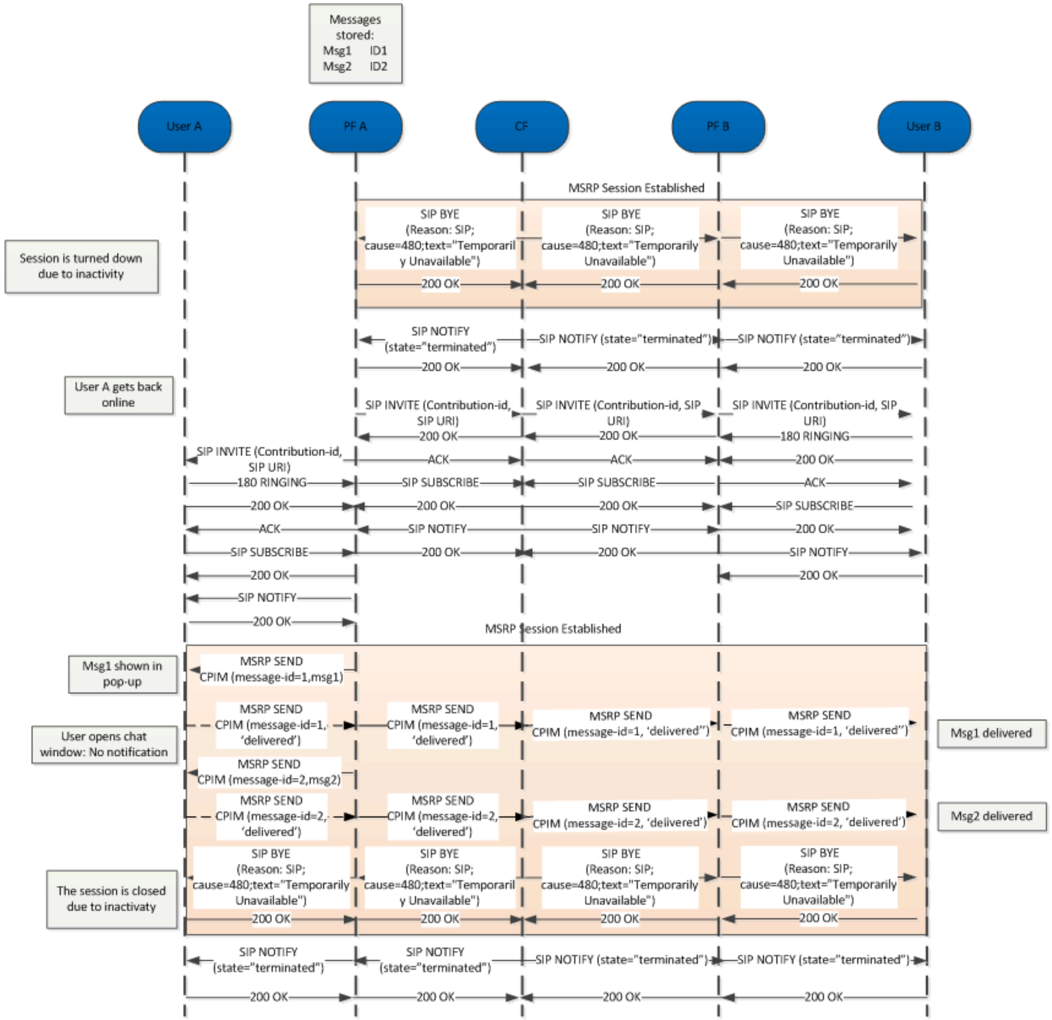

Figure 147: Deliver Group Chat Messages while Chat is idle*

*: Check NOTES 1, 15, 17, 18 and 19 in section B.1.19

B.1.18. Race Condition: user rejoins active Group Chat which is torn down due to inactivity

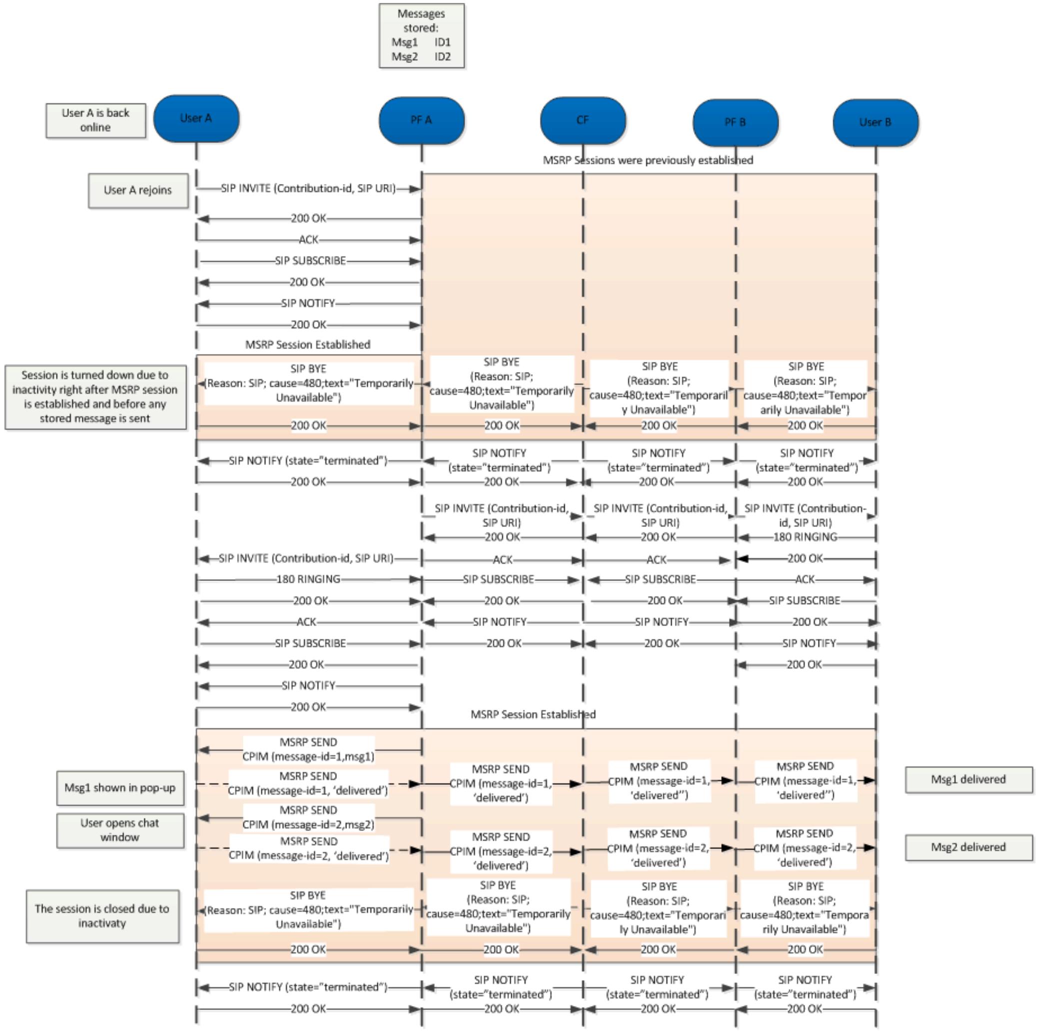

Figure 148: Rejoin in active Chat which is torn down due to inactivity*

*: Check NOTES 1, 15, 17, 18 and 19 in section B.1.19

B.1.19. Chat and store and forward diagrams: Notes

Please note the following notes apply to diagrams in section B.1:

- NOTE 1 (B.1.1, B.1.2, B.1.3, B.1.4, B.1.5, B.1.6, B.1.7, B.1.9, B.1.10, B.1.11, B.1.12, B.1.13, B.1.14, B.1.15, B.1.16, B.1.17 and B.1.18): 200 OK responses to SIP MESSAGE and MSRP SEND messages are omitted for clarity.
- NOTE 2 (B.1.3): In a multidevice scenario, a delivery notification received from User B might not end up on User A's device that sent the message. However this is not an issue, since all User A's devices will eventually receive the delivery notification upon synchronizing with the Common Message Store.
- NOTE 3 (B.1.4): See NOTE 2
- NOTE 4 (B.1.3, B.1.4 and B.1.6): B could have to handle two incoming INVITEs, one from the Messaging Server on behalf of A to deliver messages and notifications that were stored to be forwarded, and a second one directly from A who happens to want to chat with B at the same time. B should recognise the INVITE from the Messaging Server on behalf of A and not tear it down when the new INVITE directly from A arrives: The INVITE from the Messaging Server has a Referred-By header and no isfocus tag, and the INVITE directly from A does not have a Referred-By header. Please note that the same applies to the case in which the order in which the INVITEs arrive is reversed.
- NOTE 5 (B.1.3, B.1.4, B.1.5 and B.1.6): The session established by the Messaging Server to deliver deferred messages to the destination only allows the receiver (client/device) to send back notifications (that is an INVITE with referred-by header will only allow message/imdn+xml in the CPIM part). If the user replies with a new message, then a separate session shall be established (That is if User B (the receiver) wants to reply, a new INVITE should be used) after all the deferred messages have been delivered.
- NOTE 6 (B.1.2, B.1.14 and B.1.15): In the diagram we have represented one of the possible mechanisms to detect that the user is not online (wait for the 480 response), however, there are alternative mechanisms (triggers, 3rd party registration) that can be also used by the Messaging Server for the purpose.
- NOTE 7 (B.1.3, B.1.4, B.1.5, B.1.14 and B.1.15): Note that in the scenario where the MSRP socket is closed between the Messaging Server and the Terminating client (B) in a deferred message delivery (due for instance to a small connectivity loss with the PDP context remaining active) and no re-registration takes place, if there are notifications pending (delivery or displayed) and all the deferred messages have been sent to B already (no need to open a new MSRP session), SIP MESSAGE can be used to confirm the pending delivery/display notifications that could not be sent over MSRP.
- NOTE 8 (B.1.11): Note that the deferred delivery of the display notifications stored to be forwarded in the Messaging Server will be performed as shown in section B.1.6.

- NOTE 9 (B.1.11): In the absence of a Messaging Server (neither in the sender's nor in the receiver's domain) and in the case the display notification fail to be delivered because the sender is offline, these notifications will be discarded and the receiver's client does not need to retry sending them. In any case, the next time User A manages to establish a chat session with User B, all the previous messages pending to receive the displayed notification will be marked as displayed/read.
- NOTE 10 (B.1.7 and B.1.11): In those scenarios where a Messaging Server is not available, neither in the sender's nor in the receiver's network, there is a chance that display notifications carried via SIP MESSAGE may be lost if the original sending client is offline when the receiver sends those display notifications (that is the last three messages in the diagram). To overcome this limitation, a terminal or client implementation should mark all the previous messages as displayed when a new chat message is received from the receiving user.
- NOTE 11 (B.1.3, B.1.4, B.1.5 and B.1.6): The session established by the Messaging Server to deliver deferred messages or notifications should be terminated once the all the messages and notifications have been delivered. In more detail:

  - When delivering deferred messages, the session should be terminated (by sending a BYE) either (whatever is shorter) when the display notification corresponding to the last deferred message has been received by the Messaging Server or, after a timer started on the reception of the delivered notification for the last message expires. This timer is defined by the Service Provider.

- NOTE 12 (B.1.12 and B.1.13): The predefined text for accepting and leaving a session is included for illustration purposes only as it is up to the Service Provider providing the interworking to configure an appropriate an appropriate text and announce that to the SMS/MMS user when appropriate.
- NOTE 13 (B.1.13): If the SMS (or MMS) user does not respond in time, the INVITE will have timed out and the used MSISDN may even be assigned to another session. For that reason the Messaging Server should check whether the SMS (or MMS) message comes from a user that is invited to the related session and if that is not the case or the MSISDN is not assigned to any session, a message is sent back informing the user that he cannot join the session any longer.
- NOTE 14 (B.1.4, B.1.5 and B.1.6): Whether a Messaging Server sets up a session for the delivery of notifications or sends them using SIP MESSAGE requests is up to its local policy. This could depend on factors such as the number of notifications that were stored or the number of messages for which notifications can be expected (during delivery of stored messages for instance).
- NOTE 15 (B.1.1, B.1.2, B.1.3, B.1.4, B.1.5, B.1.6, B.1.7, B.1.8, B.1.9, B.1.10, B.1.11, B.1.12, B.1.13, B.1.14, B.1.15, B.1.16, B.1.17 and B.1.18): As per [RFC5438], the message-id is conveyed in the messages via the imdn.Message-ID header and in the notifications via the value of the <message-id> element in the body of the IMDN.
- NOTE 16 (B.1.12 and B.1.13): The flows show interworking with SMS, but the flows in the SIP/MSRP part of the figure also apply when interworking with MMS.

- NOTE 17 (B.1.16, B.1.17 and B.1.18): As per sections 3.4.4.1.7 and 3.4.4.3.13.
- NOTE 18 (B.1.17 and B.1.18): The flow shows the Participating Function restarting the session before attempting the delivery. This is an implementation option to ensure that a session is established when the user sends content. The Participating Function may also choose to establish this session in parallel or only when there is actual content to be sent in the Chat.
- NOTE 19 (B.1.17 and B.1.18): The flow assumes that no display notifications were requested.

B.2. Chat and store and forward diagrams with Automatic Acceptance

B.2.1. Chat without store and forward

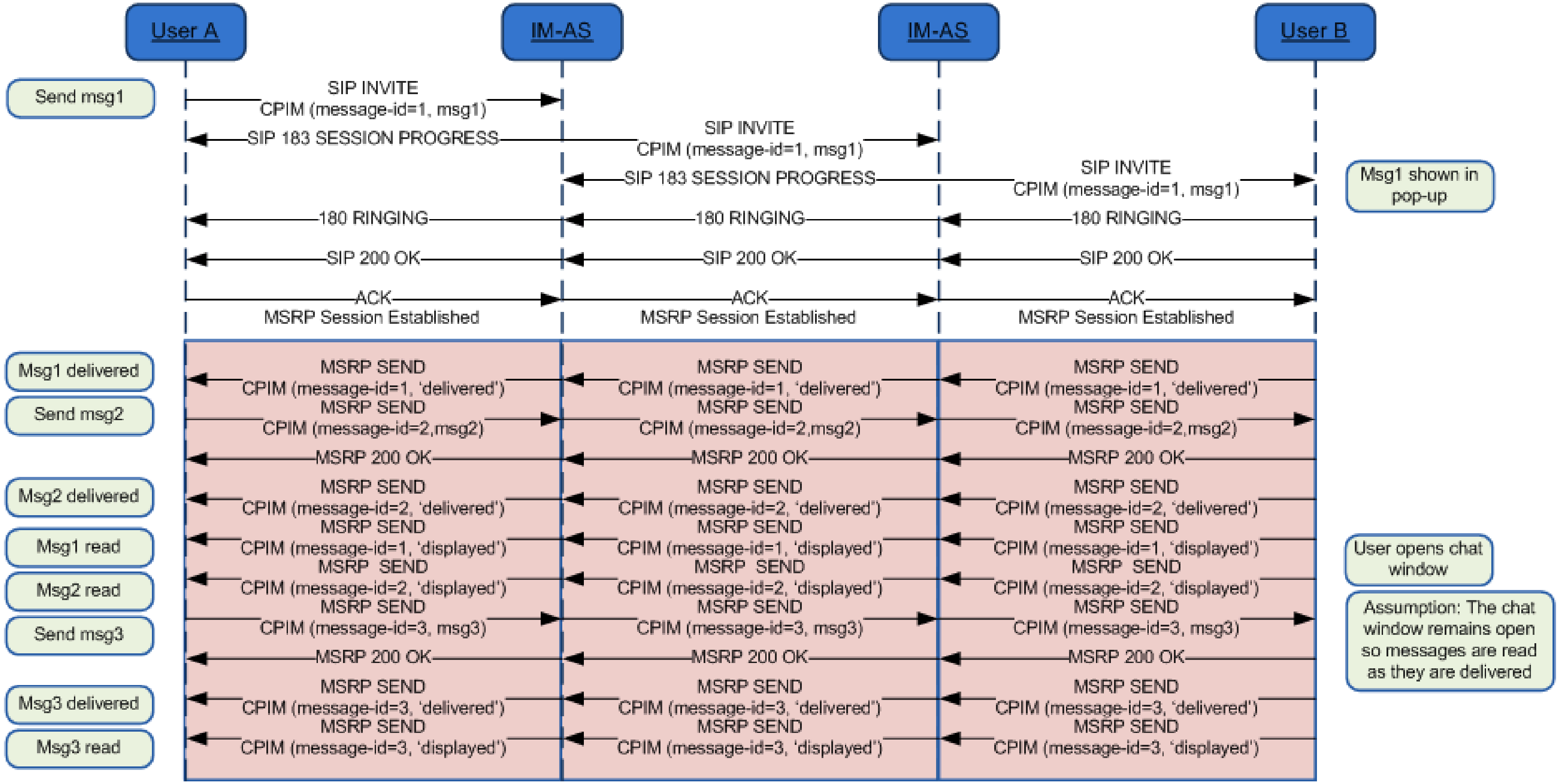

Figure 149: Chat flow without store and forward *

*: Check NOTES 1, 2, 16 and 17 in section B.2.19

B.2.2. Store and forward: Receiver offline

This case is identical to the one without automatic acceptance (see section B.1.2). NOTES 1, 2, 7 and 17 in section B.2.19 apply as well.

B.2.3. Store and forward: Message deferred delivery with sender still on an active Chat session

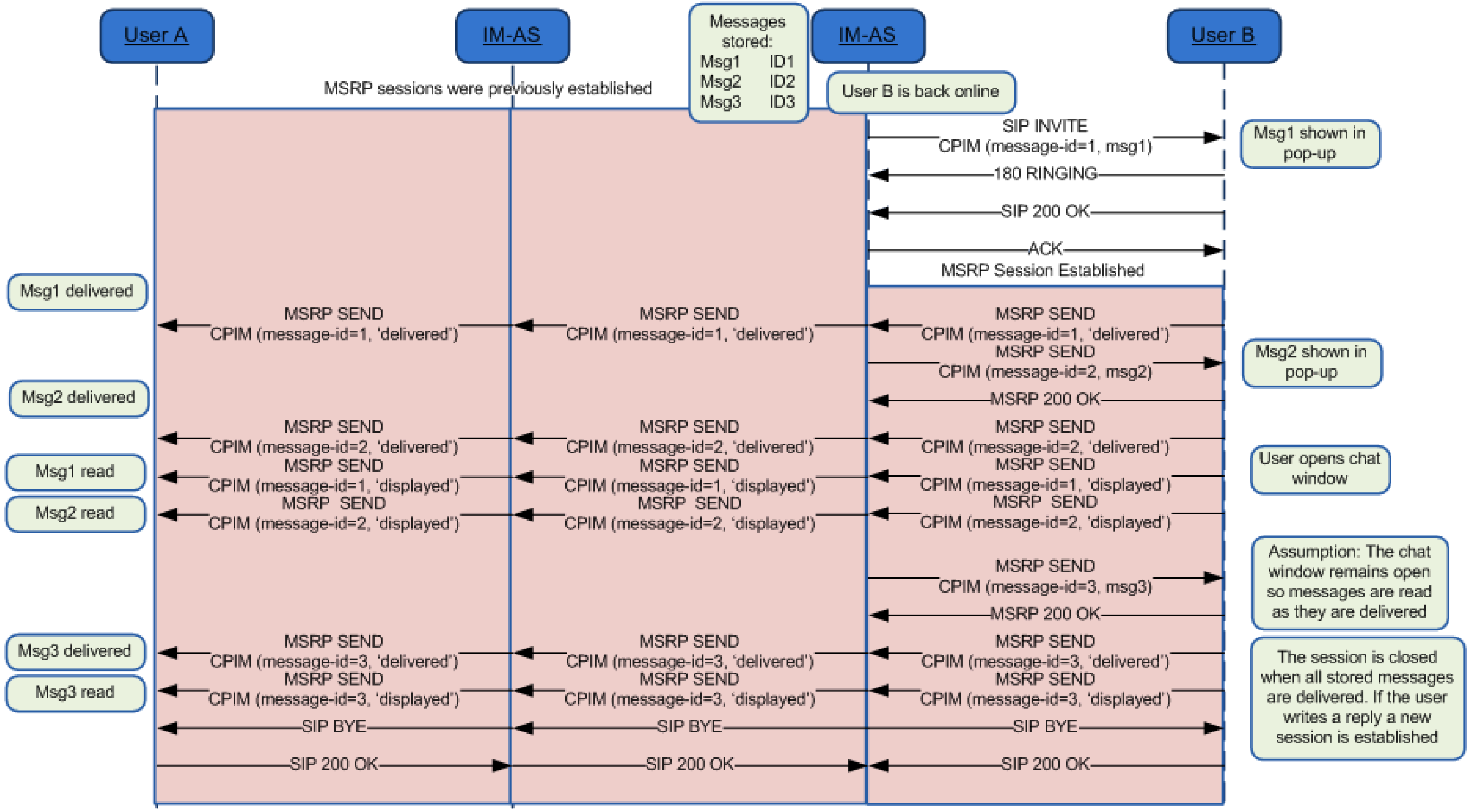

Figure 150: Store and forward: Message(s) deferred delivery with a sender still on an MSRP session*

*: Check NOTES 1, 2, 3, 5, 6, 8, 12, 16 and 17 in section B.2.19

B.2.4. Store and forward: Message deferred delivery with sender online

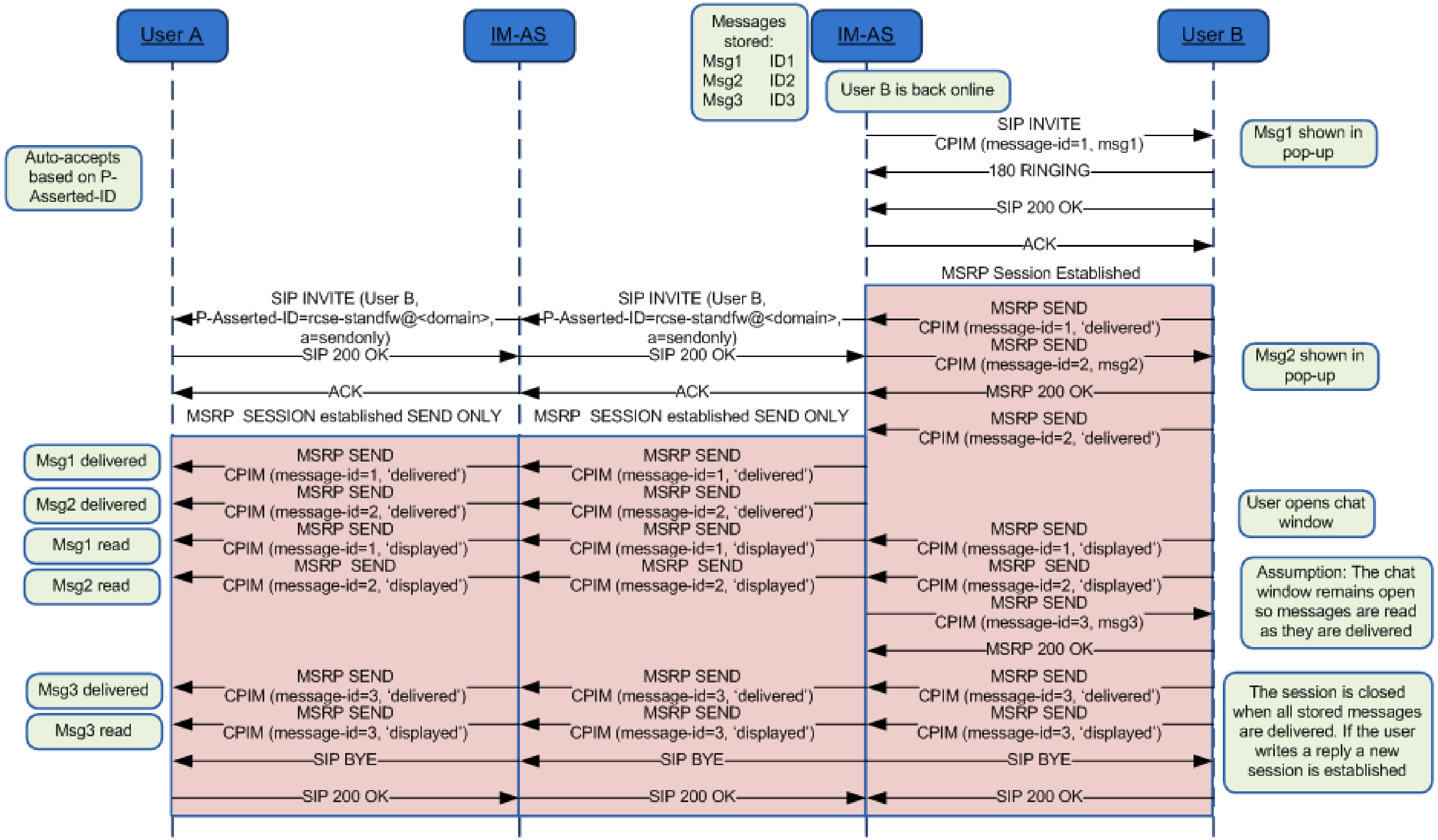

Figure 151: Store and forward: Message deferred delivery with sender online *

*: Check NOTES 1, 2, 4, 5, 6, 8, 12, 15, 16 and 17 in section B.2.19

B.2.5. Store and forward: Message deferred delivery with sender offline (delivery notifications)

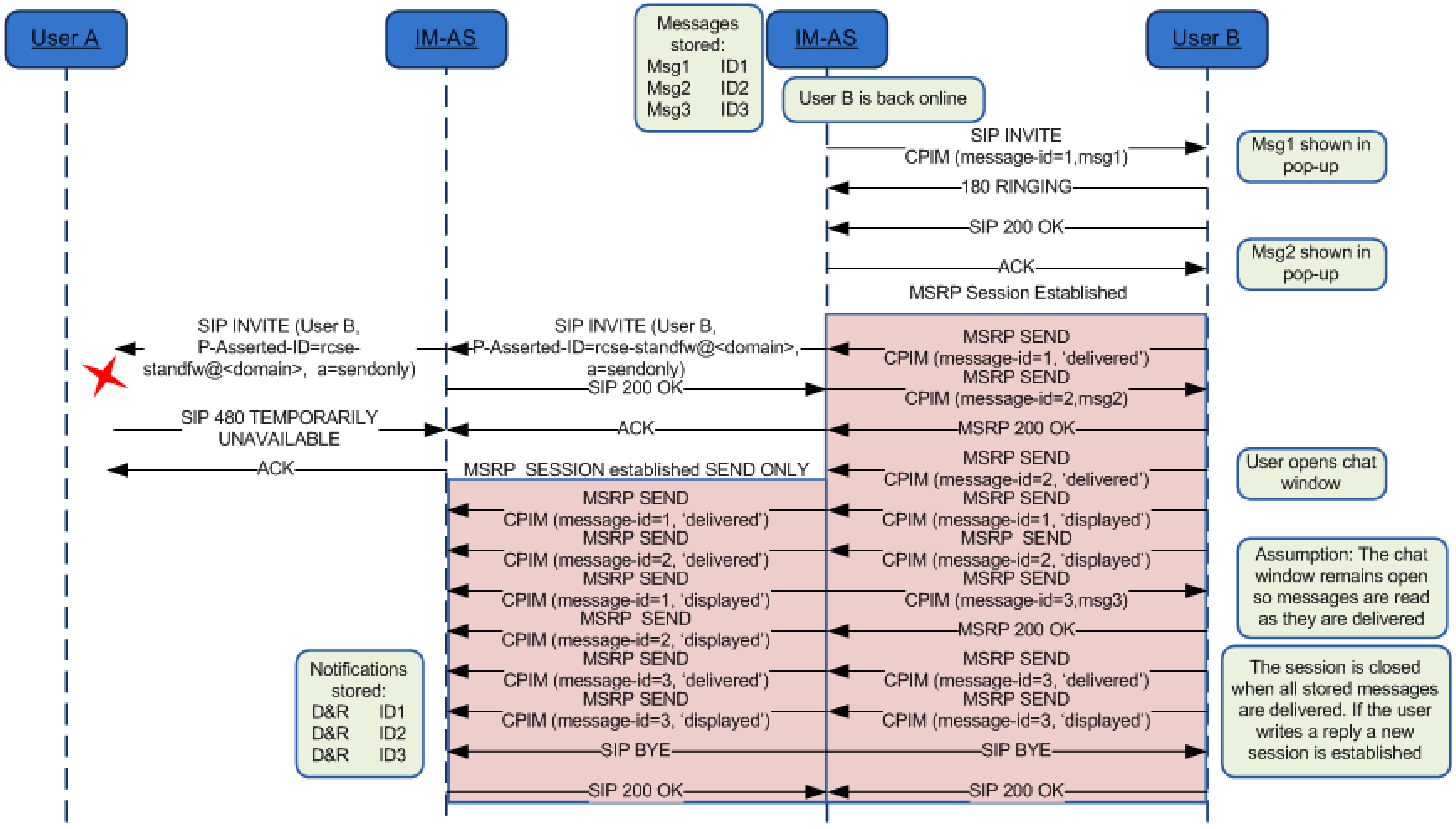

Figure 152: Store and forward: Message(s) deferred delivery with a sender offline (delivery notifications)*

*: Check NOTE 1, 2, 6, 8, 12, 15, 16 and 17 in section B.2.19

B.2.6. Store and forward: Notifications deferred delivery

This case is identical to the one without automatic acceptance (see section B.1.6). NOTES 2, 5, 6, 12, 15 and 17 in section B.2.19 apply as well.

B.2.7. Delivery of displayed notifications in an unanswered chat (without store and forward)

This case is not applicable in case of automatic acceptance.

B.2.8. Store and forward: Handling errors in the receiver's side

This case is identical to the one without automatic acceptance (see section B.1.8) taking into account NOTE 1 and 17 in section B.2.19.

NOTE: The error messages that are mapped to 486 Busy Here are listed in Table 31.

Also on the path between the IM-ASs (the Messaging Server) similar errors could occur. In that case if the originating Messaging Server supports Store and Forward, it will behave in the same way and store the message.

B.2.9. Race conditions: Simultaneous INVITEs

Even if somewhat more unlikely in case of automatic acceptance, this case is identical to the one without auto-accept (see section B.1.9). NOTES 1, 2 and 17 in section B.2.19 apply as well.

B.2.10. Race conditions: New INVITE after a session is accepted

Even if somewhat more unlikely in case of automatic acceptance, this case is identical to the one without auto-accept (see section B.1.10). NOTES 1, 2 and 17 in section B.2.19 apply as well.

B.2.11. Store and forward: Message(s) displayed notifications via SIP MESSAGE with sender offline

This case is identical to the one without automatic acceptance (see section B.1.11). NOTES 2, 9, 10, 11 and 17 in section B.2.19 apply as well.

B.2.12. Interworking to SMS/MMS with automatic acceptance at the IWF

This case is identical to the one without automatic acceptance (see section B.1.12). NOTES 1, 2, 13, 17 and 18 in section B.2.19 apply as well.

B.2.13. Interworking to SMS/MMS with manual acceptance

This case is identical to the one without automatic acceptance (see section B.1.13). NOTES 1, 2, 13, 17 and 18 in section B.2.19 apply as well

B.2.14. Message Revoke: Successful Request

This case is identical to the one without automatic acceptance (see section B.1.14). NOTES 2, 7, 8 and 17 in section B.2.19 apply as well

B.2.15. Message Revoke: Failed Request

This case is identical to the one without automatic acceptance (see section B.1.15). NOTES 2, 7, 8 and 17 in section B.2.19 apply as well

B.2.16. Re-joining a Group Chat that timed out due to inactivity

This case is identical to the one without automatic acceptance (see section B.1.16). NOTES 2, 17 and 19 in section B.2.19 apply as well

B.2.17. Deliver Group Chat Messages while Chat is idle

This case is identical to the one without automatic acceptance (see section B.1.17). NOTES 2, 17, 19, 20 and 21 in section B.2.19 apply as well.

B.2.18. Race Condition: user rejoins active Group Chat which is torn down due to inactivity

This case is identical to the one without automatic acceptance (see section B.2.18). NOTES 2, 17, 19, 20 and 21 in section B.2.19 apply as well.

B.2.19. Chat and store and forward diagrams: Notes

Please note the following notes apply to diagrams in section B.2:

- NOTE 1 (B.2.1, B.2.2, B.2.3, B.2.4, B.2.5, B.2.8, B.2.9, B.2.10, B.2.12 and B.2.13): As said in section B.2, the inclusion of the message in the INVITE request is optional. If not included, the flows would be identical, but the message would be sent in the MSRP session instead as soon as it has been established.
- NOTE 2 (B.2.1, B.2.2, B.2.3, B.2.4, B.2.5, B.2.6, B.2.9, B.2.10, B.2.11, B.2.12, B.2.13, B.2.14, B.2.15, B.2.16, B.2.17 and B.2.18): 200 OK responses to SIP MESSAGE and MSRP SEND messages are omitted for clarity.

- NOTE 3 (B.2.3): In a multidevice scenario, a delivery notification received from User B might not end up on User A's device that sent the message. However this is not an issue, since all User A's devices will eventually receive the delivery notification upon synchronising with the Common Message Store.
- NOTE 4 (B.2.4): See NOTE 3.
- NOTE 5 (B.2.3, B.2.4 and B.2.6): B could have to handle two incoming INVITEs, one from the Messaging Server on behalf of A to deliver messages and notifications that were stored to be forwarded, and a second one directly from A who happens to want to chat with B at the same time. B should recognise the INVITE from the Messaging Server on behalf of A and not tear it down when the new INVITE directly from A arrives: The INVITE from the Messaging Server has a Referred-By header and no isfocus tag, and the INVITE directly from A does not have a Referred-By header. Please note that the same applies to the case in which the order in which the INVITEs arrive is reversed.
- NOTE 6 (B.2.3, B.2.4, B.2.5 and B.2.6): The session established by the Messaging Server to deliver deferred messages to the destination only allows the receiver (client/device) to send back notifications (that is an INVITE with referred-by header will only allow message/imdn+xml in the CPIM part). If the user replies with a new message, then a separate session shall be established (That is if User B (the receiver) wants to reply, a new INVITE should be used) after all the deferred messages have been delivered.
- NOTE 7 (B.2.2, B.2.14 and B.2.15): In the diagram we have represented one of the possible mechanisms to detect that the user is not online (wait for the 480 response), however, there are alternative mechanisms (triggers, 3rd party registration) that can be also used by the Messaging Server for the purpose.
- NOTE 8 (B.2.3, B.2.4, B.2.5, B.2.14 and B.2.15): Note that in the scenario where the MSRP socket is closed between the Messaging Server and the Terminating client (B) in a deferred message delivery (due for instance to a small connectivity loss with the PDP context remaining active) and no re-registration takes place, if there are notifications pending (delivery or displayed) and all the deferred messages have been sent to B already (no need to open a new MSRP session), SIP MESSAGE can be used to confirm the pending delivery/display notifications that could not be sent over MSRP.
- NOTE 9 (B.2.11): Note that the deferred delivery of the display notifications stored to be forwarded in the Messaging Server will be performed as shown in section B.2.6.
- NOTE 10 (B.2.11): In the absence of a Messaging Server (neither in the sender's nor in the receiver's domain) and in the case the display notification fail to be delivered because the sender is offline, these notifications will be discarded and the receiver's client does not need to retry sending them. In any case, the next time User A manages to establish a chat session with User B, all the previous messages pending to receive the displayed notification will be marked as displayed/read.
- NOTE 11 (B.2.7 and B.2.11): In those scenarios where a Messaging Server is not available, neither in the sender's nor in the receiver's network, there is a chance that display notifications carried via SIP MESSAGE may be lost if the original sending client is offline when

the receiver sends those display notifications (that is the last three messages in the diagram). To overcome this limitation, a terminal or client implementation should mark all the previous messages as displayed when a new chat message is received from the receiving user.

- NOTE 12 (B.2.3, B.2.4, B.2.5 and B.2.6): The session established by the Messaging Server to deliver deferred messages or notifications should be terminated once the all the messages and notifications have been delivered. In more detail:
    - When delivering deferred messages, the session should be terminated (by sending a BYE) either (whatever is shorter) when the display notification corresponding to the last deferred message has been received by the Messaging Server or, after a timer started on the reception of the delivered notification for the last message expires. This timer is defined by the Service Provider.
- NOTE 13 (B.2.12 and B.2.13): The predefined text for accepting and leaving a session is included for illustration purposes only as it is up to the Service Provider providing the interworking to configure an appropriate an appropriate text and announce that to the SMS/MMS user when appropriate.
- NOTE 14 (B.2.13): If the SMS (or MMS) user does not respond in time, the INVITE will have timed out and the used MSISDN may even be assigned to another session. For that reason the Messaging Server should check whether the SMS (or MMS) message comes from a user that is invited to the related session and if that is not the case or the MSISDN is not assigned to any session, a message is sent back informing the user that he cannot join the session any longer.
- NOTE 15 (B.2.4, B.2.5 and B.2.6): Whether a Messaging Server sets up a session for the delivery of notifications or sends them using SIP MESSAGE requests is up to its local policy. This could depend on factors such as the number of notifications that were stored or the number of messages for which notifications can be expected (during delivery of stored messages for instance).
- NOTE 16 (B.2.1, B.2.3, B.2.4 and B.2.5): When there is automatic acceptance and the first message is carried in the initial SIP INVITE, the delivery notification may be either delivered using a SIP MESSAGE or MSRP SEND leaving the choice up to the client implementation. In the diagrams we shown before, we have followed the second option.
- NOTE 17 (B.2.1, B.2.2, B.2.3, B.2.4, B.2.5, B.2.6, B.2.9, B.2.10, B.2.11, B.2.12, B.2.13, B.2.14, B.2.15, B.2.16, B.2.17 and B.2.18): As per [RFC5438], the message-id is conveyed in the messages via the imdn.Message-ID header and in the notifications via the value of the <message-id> element in the body of the IMDN.
- NOTE 18 (B.2.12 and B.2.13): The flows show interworking with SMS, but the flows in the SIP/MSRP part of the figure also apply when interworking with MMS.
- NOTE 19 (B.2.16 and B.2.17): As per sections 3.4.4.1.7 and 3.4.4.3.13.

- NOTE 20 (B.2.17 and B.2.18): The flow shows the Participating Function restarting the session before attempting the delivery. This is an implementation option to ensure that a session is established when the user sends content. The Participating Function may also choose to establish this session in parallel or only when there is actual content to be sent in the Chat.
- NOTE 21 (B.2.17 and B.2.18): The flow assumes that no display notifications were requested.

B.3. RCS Chat and multidevice

B.3.1. Delivery prior to acceptance

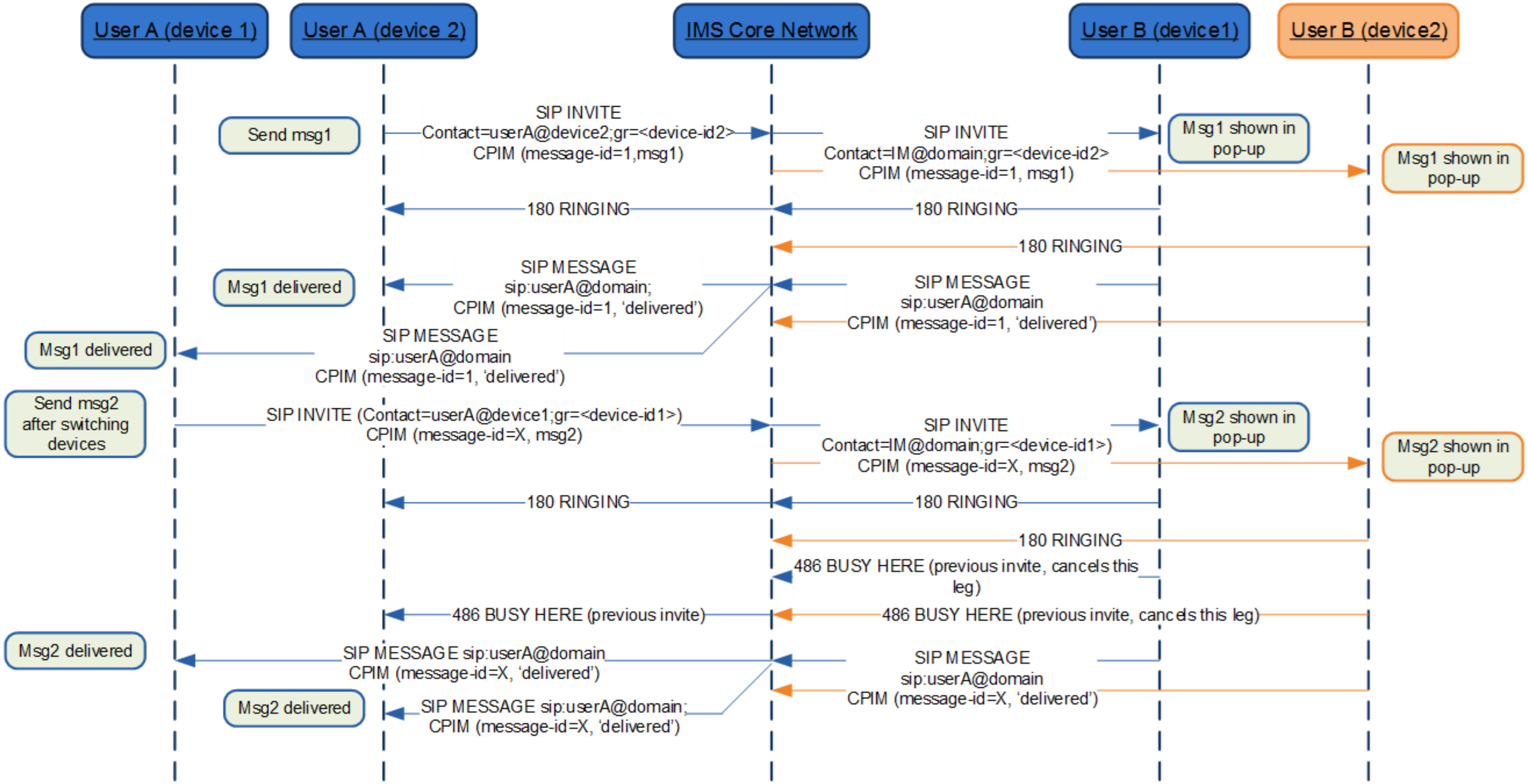

Figure 153: Chat and multidevice: Delivery prior to acceptance*

*: Check NOTES 1, 2, 3, 4 and 7 in section B.3.4

B.3.2. Post-acceptance behaviour

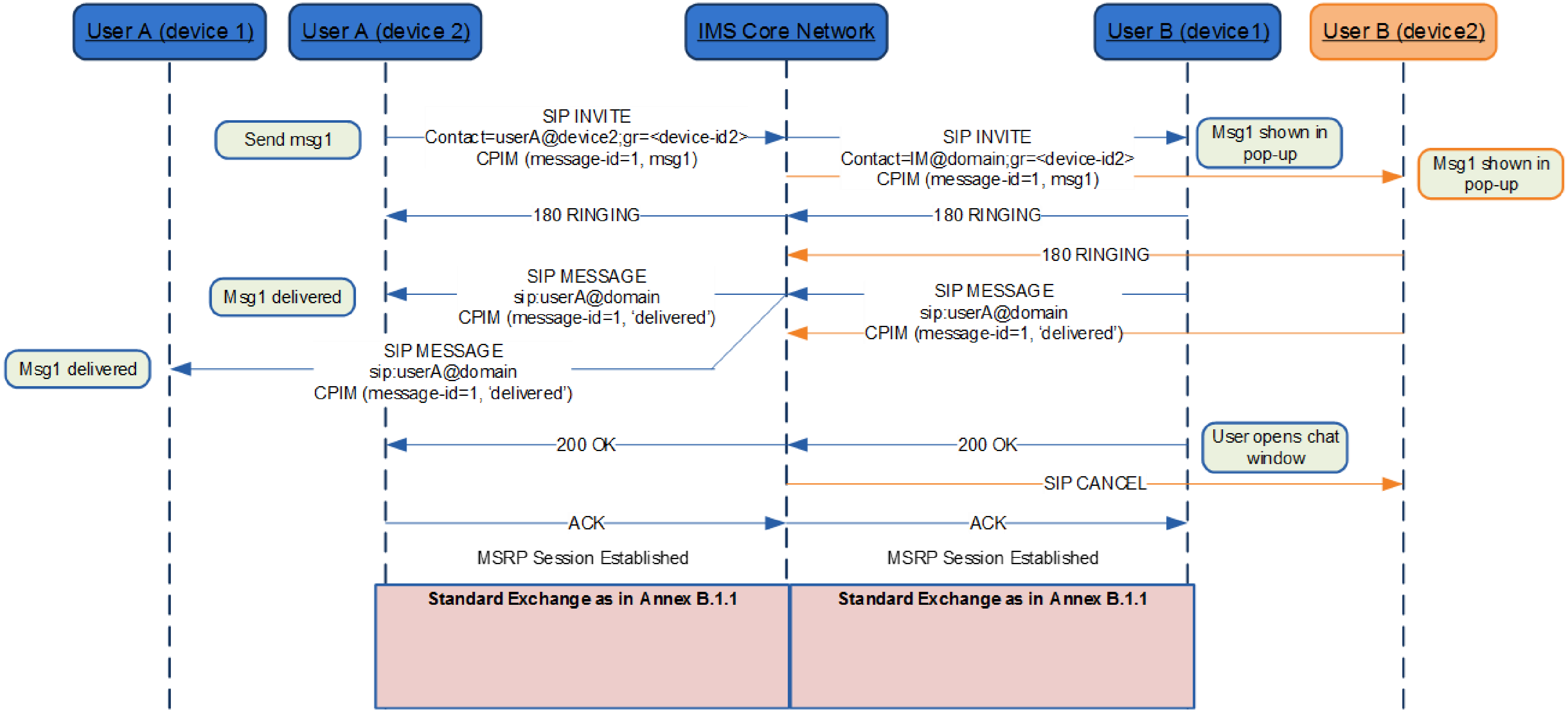

Figure 154: Chat and multidevice: Post-acceptance behaviour*

*: Check NOTES 1, 2, 3, 4 and 7 in section B.3.4

B.3.3. Behaviour with automatic acceptance

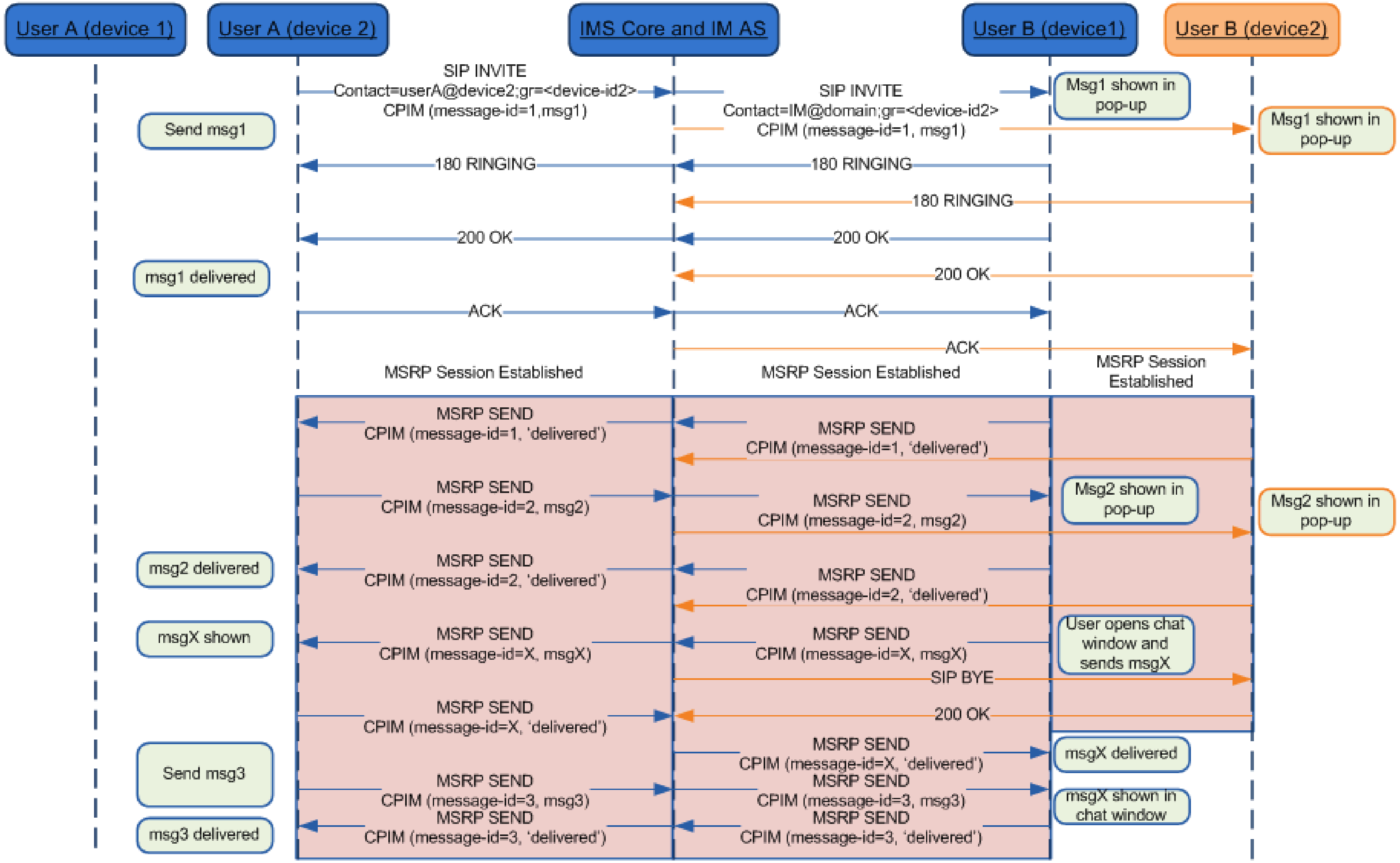

Figure 155: Chat and multidevice: Automatic acceptance*

*: Check NOTES 1, 2, 3, 4, 5, 6 and 7 in section B.3.4

B.3.4. RCS Chat and multidevice: Notes

Please note the following notes apply to diagrams in section B.3:

- NOTE 1 (B.3.1, B.3.2 and B.3.3): 200 OK responses to SIP MESSAGE and MSRP SEND messages are omitted for clarity.
- NOTE 2 (B.3.1, B.3.2 and B.3.3): The diagrams display the solution in a network supporting the pub-gruu generation. For a network supporting the sip.instance tag only, they would be equivalent with only a change of the mechanism to carry the device identifier (sip.instance instead pub-gruu).
- NOTE 3 (B.3.1, B.3.2 and B.3.3): The diagrams show that “delivered” notifications for messages for which such a notification was sent already, are suppressed by the network. As this cannot always be guaranteed, clients shall be prepared to receive such duplicate notifications and discard them silently. This holds also for display notifications and for notifications related to messages that were not sent by that client.
- NOTE 4 (B.3.1, B.3.2 and B.3.3): The SIP URIs in the diagrams (including those in the contact headers and Request URIs) are shown for illustrative purposes only. Any part of those URIs may thus differ in actual deployments. The details of the URIs are also dependent on the exact location in the network where the message is sent.
- NOTE 5 (B.3.3): The inclusion of the message in the SIP INVITE request is optional, if not supported, the message will be sent in the MSRP session as soon as that is established.
- NOTE 6 (B.3.3): To support this case forking in the terminating side needs to be done at the Messaging Server using the mechanisms defined in section 2.11.2 as forking in the IMS core will lead to a race condition.
- NOTE 7 (B.3.1, B.3.2 and B.3.3): As per [RFC5438], the message-id is conveyed in the messages via the imdn.Message-ID header and in the notifications via the value of the <message-id> element in the body of the IMDN.

B.4. Common Message Store Interaction: IMAP Flows (informative)

B.4.1. Summary of Use Cases

The following use cases are covered:

- Use Case 1: Device connects to the IMAP server
- Use Case 2: Device checks for new content
- Use Case 3: Device stores an SMS
- Use Case 4: Device deletes a specific conversation
- Use Case 5: Device saves a message to archive
- Use Case 6: Device checks for new content when server does not support XLIST-CHANGEDSINCE (for backward compatibility)

For the purpose of the Use Case examples described in this annex, the following assumptions are made for the initial view of the Common Message Store:

- RCS user A has several conversation folders under the Default folder
- Each Conversation folder contains all objects related to well-known criteria,
    - e.g. “TEL-URI-of-B”, or “Conversation-ID-for-GC-ABCD”
- The “Default/TEL-URI-of-B” folder contains:
    - 1 session for a 1-to-1 chat between A&B, with 5 objects:
        - the 1st message from the INVITE sent by this user
        - 1 received IMDN (display notification) for that message
        - 1 received reply message
        - The Session Info Object (SIO) for this conversation[42]
- The “Default/TEL-URI-of-C” folder contains:
    - 2 standalone (pager mode) interworked SMS messages
        - 1 sent to C, 1 received from C
    - 1 new 1-to-1 chat message from C:
        - the 1st message from the INVITE received
        - An SIO for this conversation[42]
        - 1 new IMDN (display notification) to C, sent using SIP MESSAGE
- The “Default/Conversation-ID-for-GC-ABCD/Contribution-ID-for-GC-ABCD” session history folder contains:
- 1 SIO for the Group Chat (GC) containing subject “Lunch?”

[42] the SIO only needs to be stored once per conversation since it provides information not otherwise in the MSRP chat messages

- the GC was started by this user, between A, B, C, & D

- 1 message sent by this user, for which user had requested a display notification
- 1 message received by this user from B
- 1 Group State Object (GSO), because the GC has ended, containing the participant list at the end of the GC
- 1 new IMDN (display notification) from D, received as a SIP MESSAGE

    - Received while device was offline.

NOTE1: As per OMA CPM, the Group Chat session history folder is labelled with the Contribution-ID. As defined in this specification Contribution-ID has the same value as the Conversation-ID. Hence both session history sub folder and conversation folder have the same name.

NOTE2: In the Common Message Store example view for user A, IMDN display notifications are shown for messages when appropriate. However, to simplify the use cases, IMDN delivery notifications are not shown in the figures nor in the example use case flows.

Logical view for user A

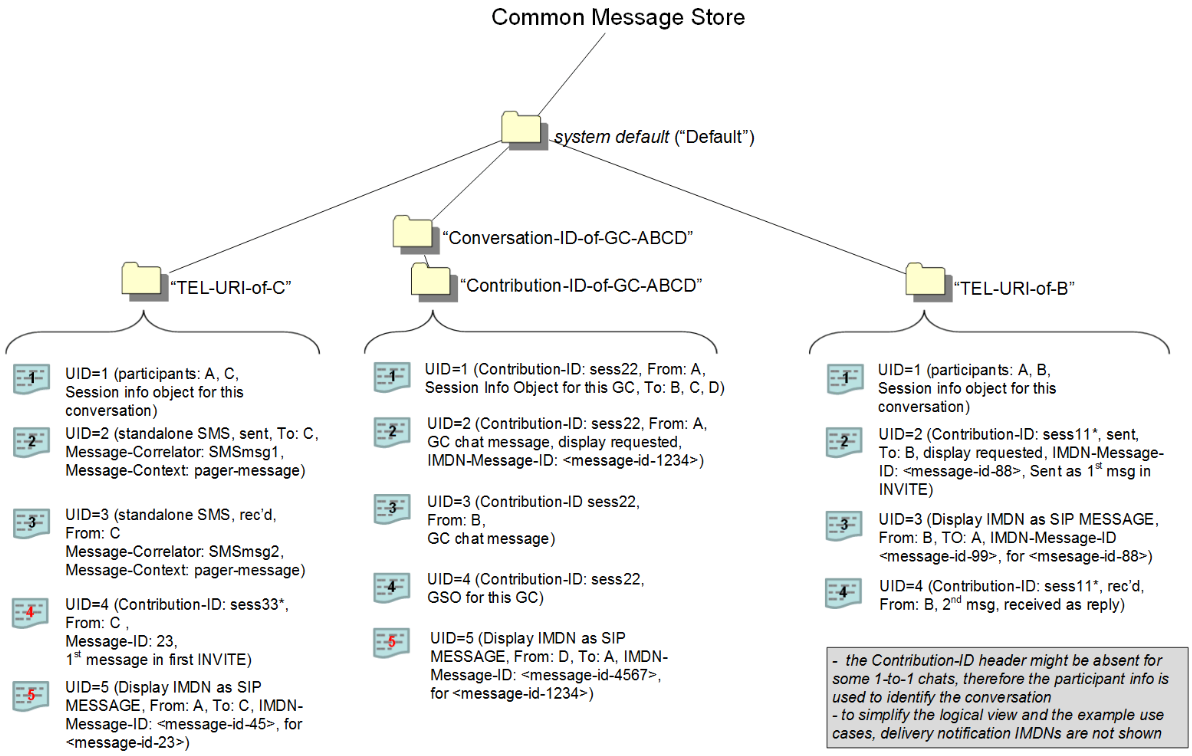

Figure 156: Initial Common Message Store view for IMAP synchronization flows

B.4.2. Use Case 1: Device connects to IMAP server

- Preconditions:

    - No connection with IMAP server

- High level flow:
  - The device connects to the server using the URL configured in the MESSAGE STORE URL attribute, starts the TLS layer, checks capabilities, and logs into the user's Message Store account using the MESSAGE STORE USER and PASSWORD configuration attributes or the bootstrapped security association.
    - Connect
    - Check capabilities
    - Login
    - Check capabilities after login

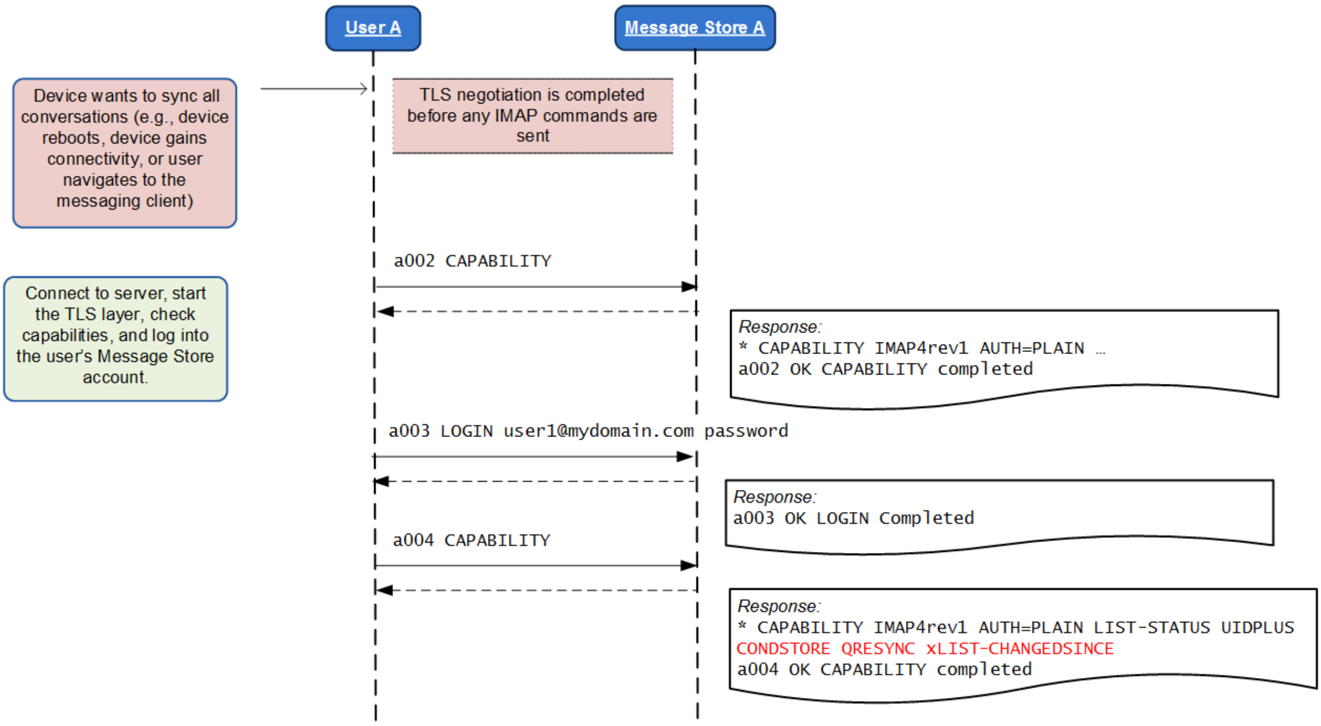

Figure 157: Use Case 1: Device connects to IMAP server

B.4.3. Use Case 2: Device checks for new content

- Preconditions:
  - Initial Common Message Store view, with 1 new object and 2 objects with changed flags in each of the TEL-URI-of-C folder and the group chat Conversation-ID-of-GC-ABCD folder.
  - Because the device was off or regains connectivity, or for a user with multiple devices, because the user has just picked up this device and navigated into the messaging application, the device needs to do a complete re-synchronisation to check for new content
  - The device has already connected to the server
  - The client has saved the highest MODSEQ value for all mailboxes from the last synchronisation
- High level flow:

- List folders (with LIST-STATUS extension described in [RFC5819], CONDSTORE extension described in [RFC7162] and XLIST-CHANGEDSINCE as described in [RCS-CPM-MSGSTOR-ENDORS])

  - The LIST command contains the argument CHANGEDSINCE
  - Although STATUS could still be used, it is no longer needed. The returned list only includes those mailboxes with MODSEQ larger than the value provided
  - For the Default folder, there is no need to search for expunged messages since expunged messages shall remain part of the local folder anyway
  - The LIST response shows folders used in the example. The returned list includes only those folders with MODSEQ larger than the value provided, which are the only ones of interest to the client
  - In this case, in the Default folder the device notices new messages/changed flags messages that one of the messages is archived
  - Since the HIGHESTMODSEQ values for each of the "Default/TEL-URI-of-C" and the "Default/Conv-with-GC-ABCD/Conv-with-GC-ABCD" folders are different from what the device has stored, a synchronization is needed on both folders

    - For each folder:

NOTE: the device decides which folder(s) to synchronize first, based on e.g., most recent user activity.

- Select the folder using CONDSTORE
- Fetch flags and MIME headers for all UIDs newer than the last remembered UID
- Compare meta information, e.g. IMDN-Message-IDs for 1-to-1 Chat, with messages received via SIP/MSRP
- Discover new objects (also including those received over SIP so a UID can be associated with each "known" message)
- Fetch completely each relevant new object so a UID can be associated with each "known" message , being sure to fetch it in a way so that it remains unread (unseen) if it has not yet been read/seen by the user (i.e. using BODY.PEEK)
- Fetch each object with changed flags using CHANGEDSINCE (1 to previous max UID).
- Arrange them in appropriate conversational views
- Apply any IMDNs received for sent messages
- Apply any \Seen flags and \Deleted flags and Archived flags received
  - Show new received messages as read when they have the \Seen flag set
  - No longer show in the conversational view any messages that have the \Deleted flag set

  - Optionally show archived messages as archived

- Log out, disconnect since there are no further folders to synchronise in this example

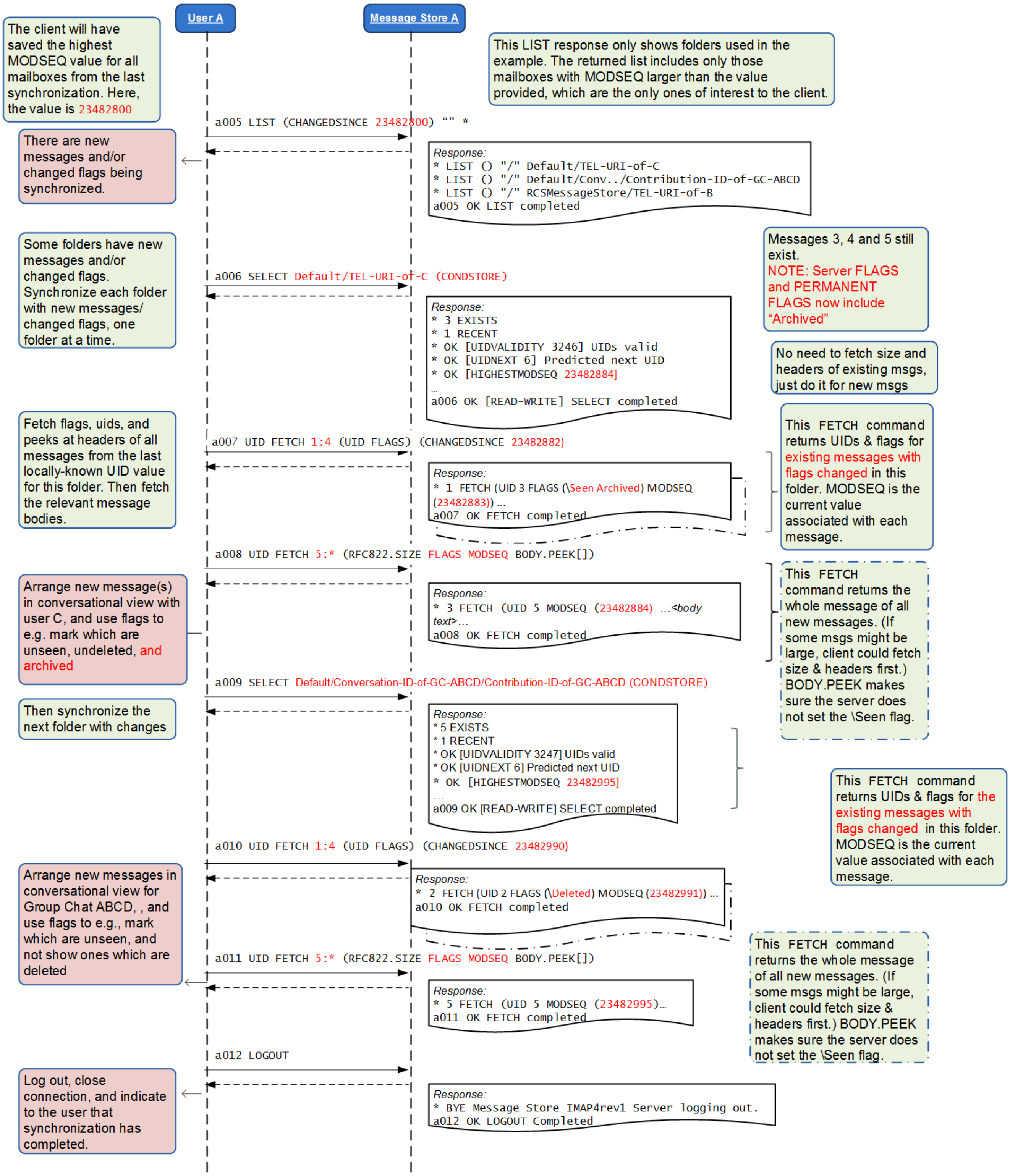

Figure 158: Use Case 2: Device checks for new content

B.4.4. Use Case 3: Device stores an SMS

- Preconditions:

- Same initial Common Message Store view, with 3 new objects
- The device has already connected to the server
- The device knows that this conversation is between users A and B
    - Normally, it is not necessary to fetch all objects related to a conversation from the Common Message Store
- The device has not found any standalone messages which match the To/From/Message-Correlator of the received SMS
    - The device has also performed a synchronization (Use Case 2) for the particular conversation this SMS belongs to thus it can be sure that the received SMS doesn’t match anything in the Common Message Store either

- High level flow:
    - Select the appropriate conversation folder, e.g. “Default/TEL-URI-of-B”
    - Append the SMS message to the folder
        - Message-Context header is set to “pager-message”
        - If the SMS MESSAGE STORE configuration parameter is set to 1, the device shall set the Message-Correlator header based on the content of the first part of the message, as defined in section 4.1.4.4
        - If the SMS MESSAGE STORE configuration parameter is set to 2, the device shall not set the Message-Correlator header
        - Content-Type of the body is set to Message/CPIM
    - The device either logs out or selects another folder

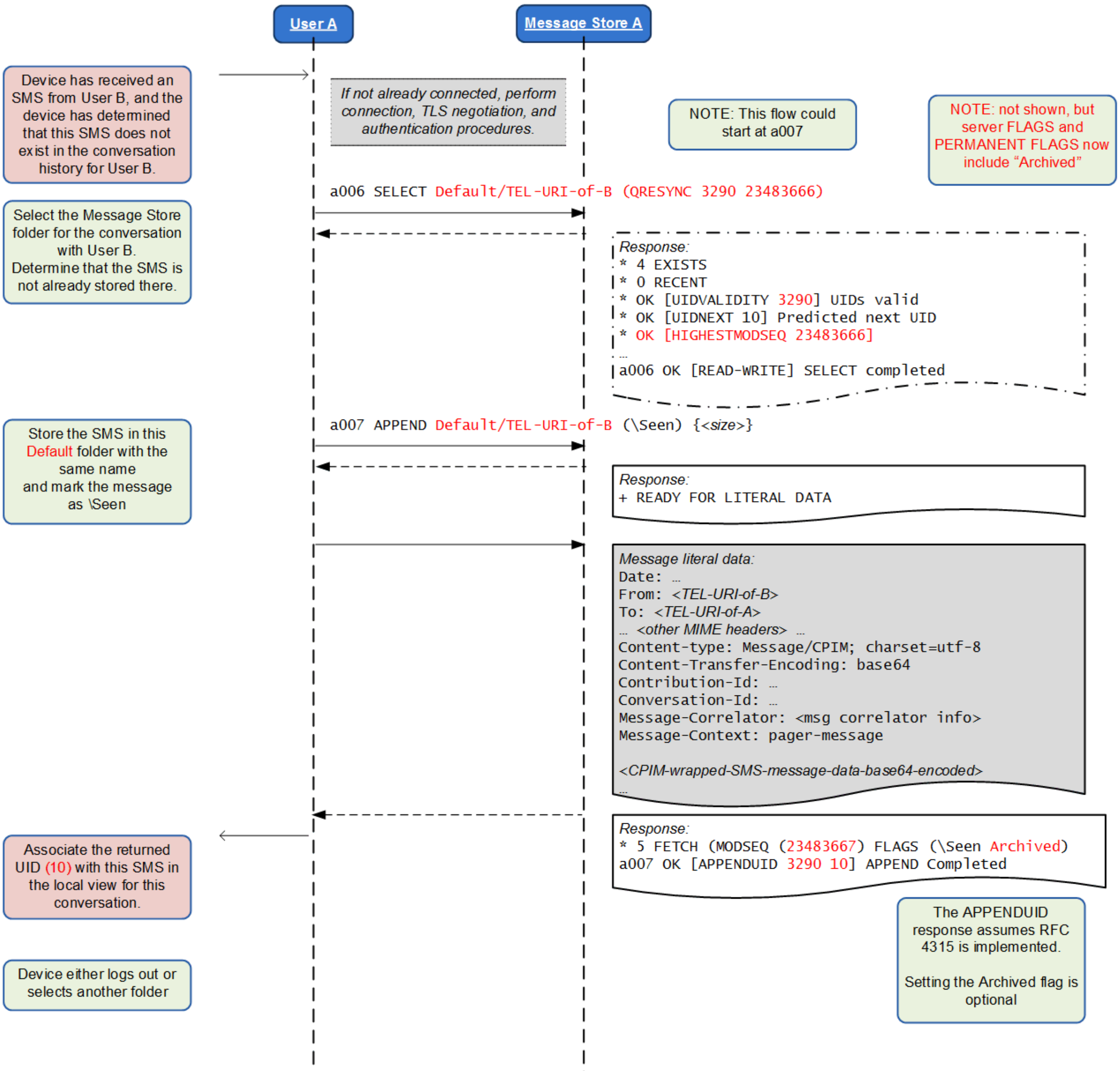

Figure 159: Use Case 3: Device stores an SMS

B.4.5. Use Case 4: Device deletes the conversation between A and B

- Preconditions:
  - Same initial Common Message Store view, all messages have been synchronised (Use Case 2)
  - Device knows that this conversation is between users A and B
  - Device knows the individual UIDs for each message and associated IMDNs between A and B

NOTE: Care should be taken to warn the user about deleting messages not yet seen

  - It is left to device implementation how to present deleted messages
  - The device has already connected to the server

- High level flow:

- Under the Default folder, set the \Deleted flag on all messages to be deleted
  - This can be used to delete all messages in a conversation folder under the Default folder, or only a subset of messages, as desired. The conversation folder itself is not deleted, but will be removed automatically on expiry of all messages in the folder.
  - Use FLAGS instead of +FLAGS, and include all other existing flags except Archived, so that a message will never have both \Deleted and Archived flags set at the same time
- Optionally log out

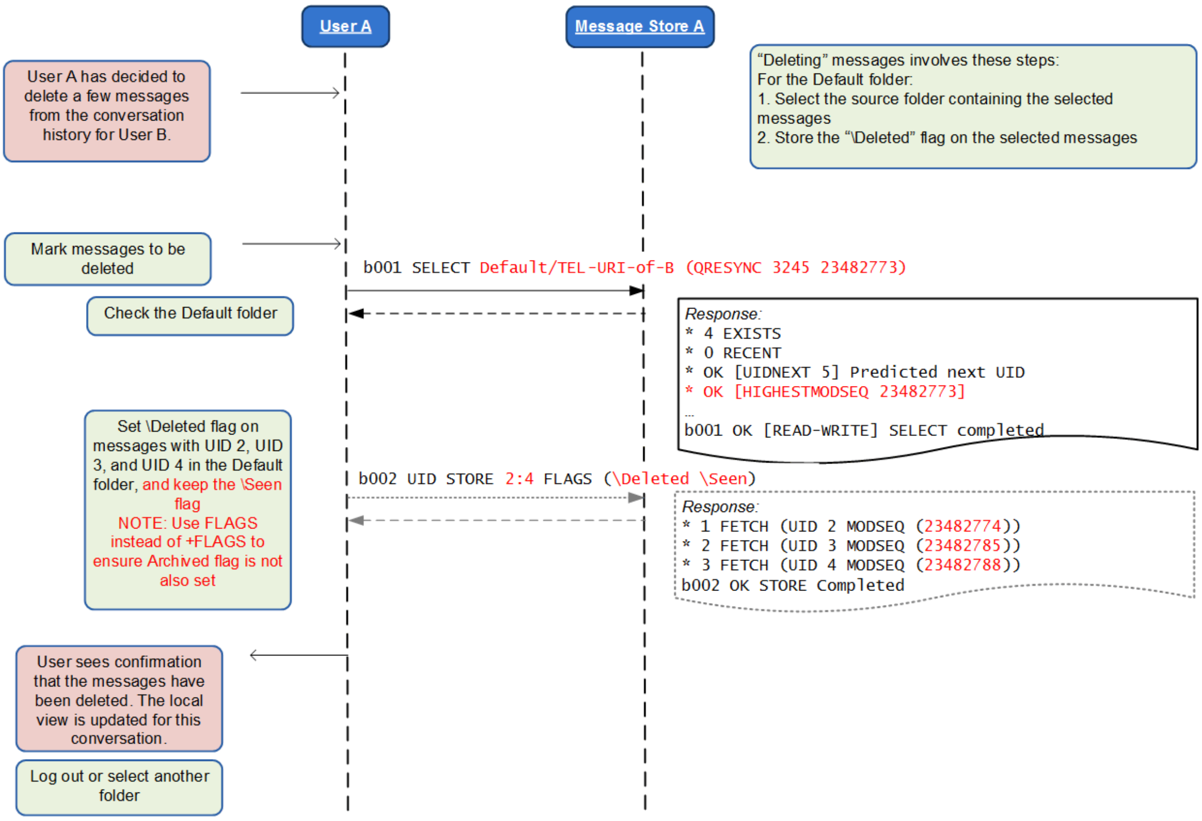

Figure 160: Use Case 4: Device deletes messages from a conversation between A and B by setting the \Deleted flag

B.4.6. Use Case 5: Device saves a message to archive

- Preconditions:
  - Same initial Common Message Store view, all messages have been synchronised (Use Case 2)
  - Device knows that this conversation is between users A and B
    - Device knows the individual UIDs for each message and associated IMDNs between A and B
    - If it does NOT know the UIDs, the device must first fetch all related messages and associated IMDNs

- Normally, it is not necessary to fetch all objects related to a conversation from the Common Message Store

- It is left to device implementation how to present messages from the archive
- The device has already connected to the server

- High level flow:

- Select the appropriate conversation folder, e.g. "Default/TEL-URI-of-B"
- Store Archived flag on the specific message(s) in the Default folder to be archived
- Device either logs out or selects another folder

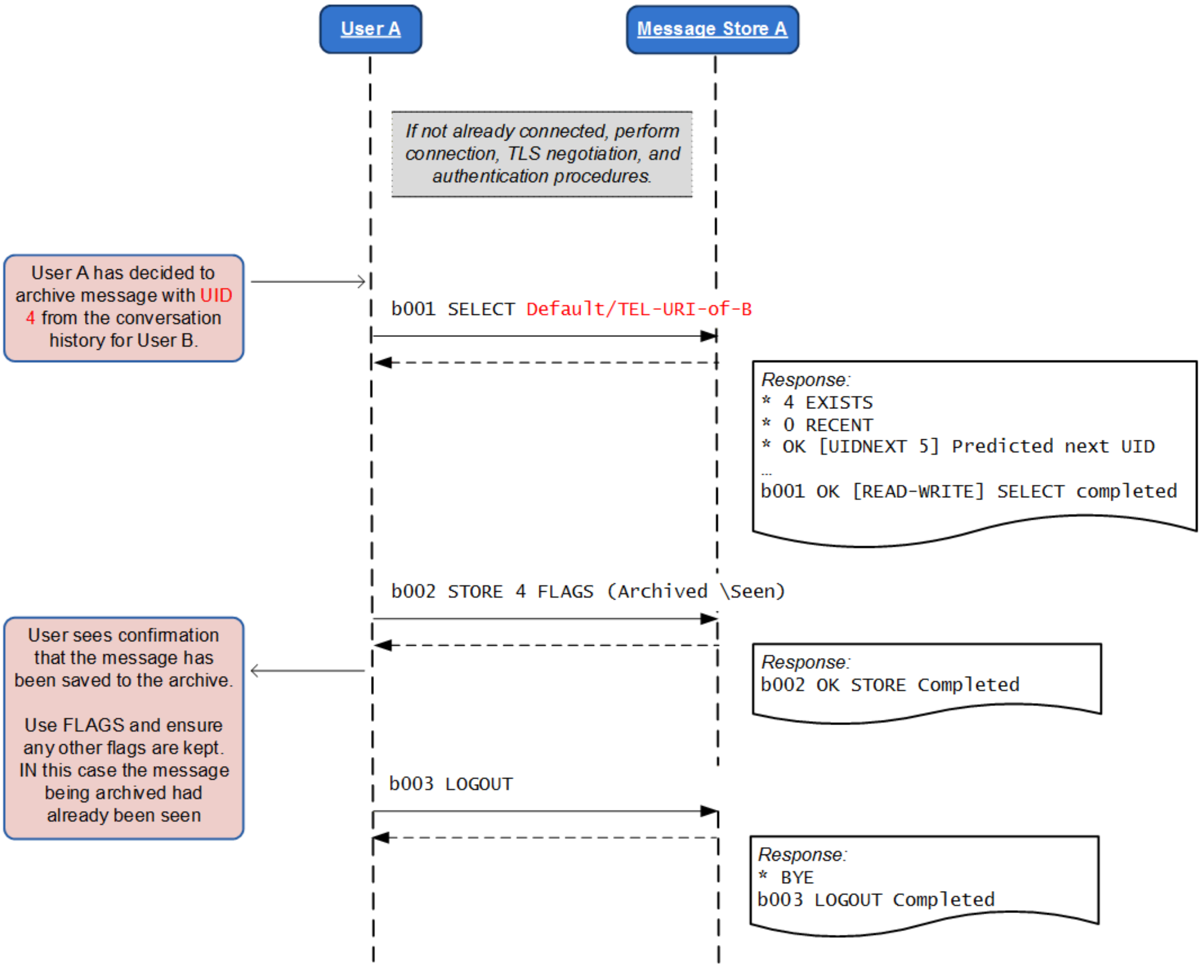

Figure 161: Use Case 5: Device archives a message

B.4.7. Use Case 6: Device checks for new content when server does not support XLIST-CHANGEDSINCE (for backward compatibility)

- Preconditions:

- Initial Common Message Store view, with 1 new object and 2 objects with changed flags in each of the TEL-URI-of-C folder and the group chat Conversation-ID-of-GC-ABCD folder.

    - Because the device was off or regains connectivity, or for a user with multiple devices, because the user has just picked up this device and navigated into the messaging application, the device needs to do a complete re-synchronisation to check for new content
    - the device has already connected to the server

- High level flow:

    - List folders (with LIST-STATUS extension described in [RFC5819] and CONDSTORE extension described in [RFC7162])

        - HIGHESTMODSEQ and UIDVALIDITY will indicate if anything new has arrived
        - If HIGHESTMODSEQ is higher and UIDVALIDITY is the same, new messages have arrived and/or flags have changed and the folder needs to be synchronised
        - If UIDVALIDITY has changed the entire folder needs to be re-synchronised based on the server content
        - For the Default folder, there is no need to search for expunged messages since expunged messages shall remain part of the local folder anyway, so CONDSTORE is used
        - In this case, in the Default folder the device notices new messages/changed flags
        - Since the HIGHESTMODSEQ values for each of the "Default/TEL-URI-of-C" and the "Default/Conv-with-GC-ABCD/Conv-with-GC-ABCD" folders are different from what the device has stored, a synchronisation is needed on both folders

            - For each folder:

NOTE: the device decides which folder(s) to synchronise first, based on e.g., most recent user activity.

                - Select the folder using CONDSTORE
                - Fetch flags and MIME headers for all UIDs newer than the last remembered UID
                - Compare meta information, e.g. IMDN-Message-IDs for 1-to-1 Chat, with messages received via SIP/MSRP
                - Discover new objects (also including those received over SIP so a UID can be associated with each "known" message)
                - Fetch completely each relevant new object so a UID can be associated with each "known" message , being sure to fetch it in a way so that it remains unread (unseen) if it has not yet been read/seen by the user (i.e. using BODY.PEEK)
                - Fetch each object with changed flags using CHANGEDSINCE (1 to previous max UID).
                - Arrange them in appropriate conversational views
                - Apply any IMDNs received for sent messages

        - Apply any \Seen flags and \Deleted flags received
            - Only show new received messages as unread when they do not have the \Seen flag set
            - No longer show in the conversational view any messages that have the \Deleted flag set
    - Log out, disconnect since there are no further folders to synchronise in this example

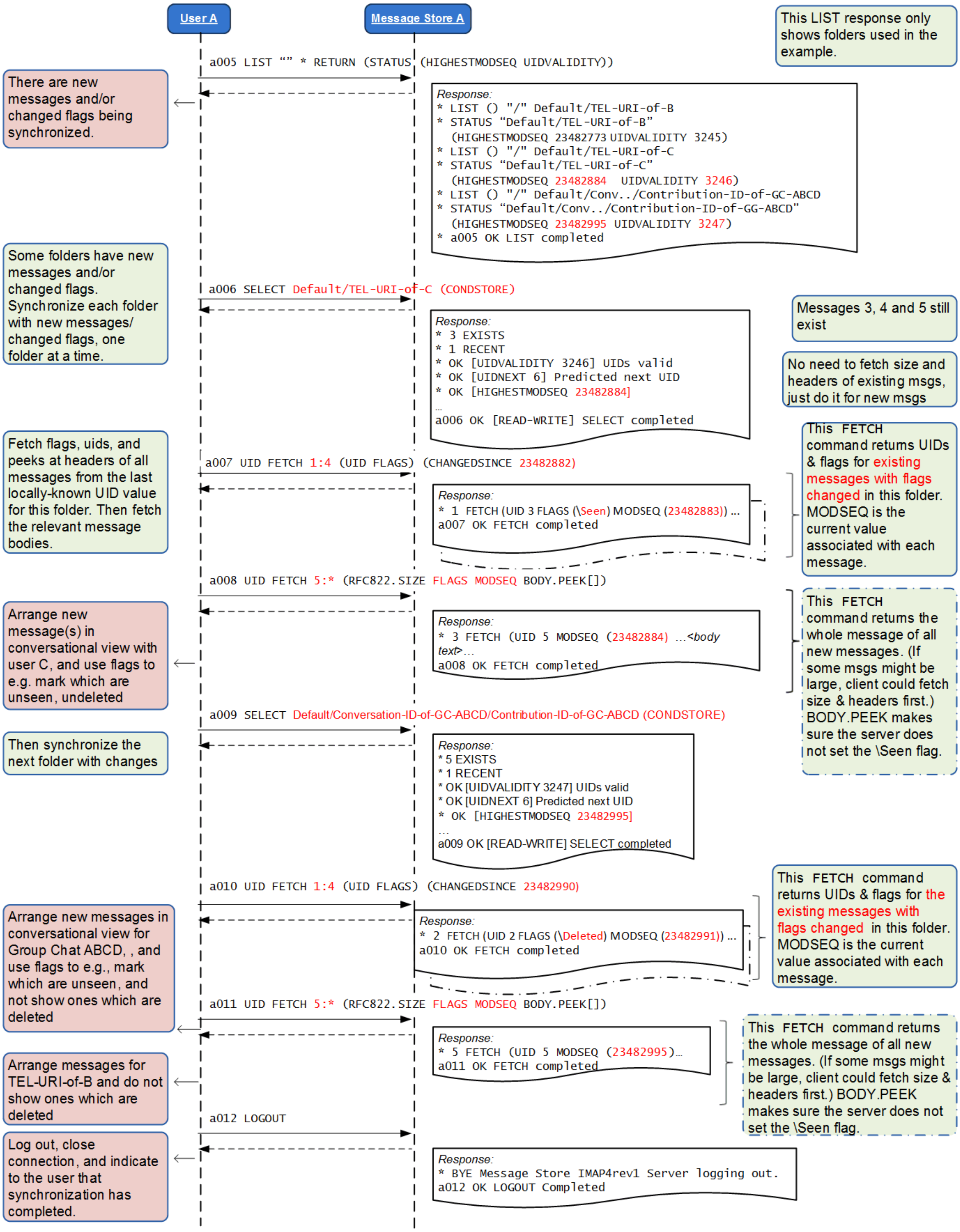

Figure 162: Use Case 7: Device checks for new content when server does not support XLIST-CHANGEDSINCE (for backward compatibility)

Annex C Special Procedures

C.1. SIP/TCP and NAT traversal

As specified in section 2.8 when using SIP over TCP (or TLS), the client shall rely on the CRLF mechanism defined in [RFC6223]. However [RFC6223] does not provide the means to negotiate the direction in which these keep-alive requests are sent (it's always the party that initiated the SIP request that has to send keep-alive requests) and a device OS's scheduling policy may not always allow the client to meet the timing requirements for sending keep-alive requests. To overcome these limitations for clients running on such platforms a mechanism is provided in this annex which is also specified in an internet draft that has been submitted to the IETF (see [IETF-DRAFT-RKEEP]). This mechanism allows these clients to request to reverse the direction in which the keep-alive requests are sent (that is they will be sent from network to client) by including an 'rkeep' parameter in the Via header of the SIP request that is used in the same way as the 'keep' parameter defined in [RFC6223].

Like the server in [RFC6223], the client may include a proposed frequency (in seconds) of the keep-alive period by adding a value to the '*rkeep'* parameter (e.g. "rkeep=600"). This frequency shall not be set to a value smaller than 30 seconds. An Edge Proxy supporting this mechanism that receives requests that contain an 'rkeep' parameter in the top-most Via header can provide the following responses:

- If the *rkeep* value is provided by the client (e.g. rkeep=600) and it is acceptable according to the service provider policies, the registration response shall include the '*rkeep*' parameter in the top-most Via header when sending a reliable response on that request and shall remove the value (i.e. *rkeep* is sent back without a value).
- If the rkeep value is provided by the client but it is not acceptable based on the Service Provider policies, the Edge proxy shall include the '*rkeep*' parameter in the top-most Via header when sending a reliable response on that request and shall set the value to a default one (i.e. rkeep=180 [assuming 180 is the default value]).
- If the rkeep value is not provided by the client (e.g. rkeep without an specified value), the Edge Proxy shall provide a frequency value by setting a default value to the 'rkeep' parameter in its response (i.e. rkeep=180 [assuming 180 is the default value]).

Then it shall send double CRLF "ping" requests as defined in [RFC5626] to the client thereby complying to the specified interval and considering the connection as failed when no single CRLF "pong" response is received within 10 seconds.

An Edge proxy not supporting this mechanism shall not modify the rkeep parameter included by the client. The fact the value introduced by the client is not modified by the Edge Proxy shall be interpreted by the client as the Edge Proxy does not support the network initiated keep alive. Please note that this approach guarantees backwards compatibility.

NOTE1: It is highly recommended that clients not experiencing such scheduling limitations use the standard 'keep' mechanism defined in [RFC6223] and send the keep-alive requests themselves. For those clients the implementation of this section is therefore optional.

NOTE2: Alternatively a Service Provider could decide to rely on client platform specific notification mechanisms

NOTE3: The requirement to extend the keep-alive procedures to support network-initiated keep-alives has been brought into the IETF for standardisation (see [IETF-DRAFT-RKEEP]). The procedures here will be updated once that work is completed. In particular this standardisation process should allow the client to detect that the network does not support network-initiated keep-alives as described above.

C.2. Errata for RFC 5438

The following errata have been reported for [RFC5438] in [RFC5438Errata] and is important to be taken into consideration for RCS with respect to messaging and chat services:

- Errata ID: 3013
- Status: Held for Document Update
- Type: Technical
- Reported By: Dan Price
- Date Reported: 2011-11-04
- Held for Document Update by: Robert Sparks
- Section 7.2.1.1 says:

  From: Bob <im:bob@example.com>
  To: Alice <im:alice@example.com>
  NS: imdn <urn:ietf:params:imdn>
  imdn.Message-ID: d834jied93rf
  Content-type: message/imdn+xml
  Content-Disposition: notification
  Content-length: ...

- It should say:

  From: Bob <im:bob@example.com>
  To: Alice <im:alice@example.com>
  NS: imdn <urn:ietf:params:imdn>
  imdn.Message-ID: d834jied93rf

  Content-type: message/imdn+xml
  Content-Disposition: notification
  Content-length: ...

- Notes:

  None of the examples in this RFC (Request For Comments) comply with the format of CPIM defined in RFC 3862, in which the message metadata headers are separated from the headers of the encapsulated MIME object by a blank line.

C.3. Definition of RCS related MIME headers

NOTE: ABNF definitions of Contribution-ID, Conversation-ID and Message-Correlator can be found via [RCS-CPM-MSGSTOR-ENDORS].

C.3.1. Definition of Message-Direction

The Message-Direction header indicates whether the message in the Common Message Store has been originally sent or received by the user.

The header is defined as an extension to the [RFC5322] field definitions. The limits for the occurrence of the field are defined in the following table:

Field	Min Number	Max Number
Message-Direction	0	1

Table 224: Message-Direction header

The field itself is defined in ABNF as follows:

```
message-direction = "Message-Direction:" message-direction-part CRLF
message-direction-part = message-direction-value / extension
message-direction-value = "sent" / "received"
extension    = *CHAR
```

Example:

```
Message-Direction: sent
```

A client receiving the Message-Direction header with a value other than "sent" or "received" shall ignore the header.

C.3.2. Definition of Replace-Short-Message-Type

The Replace-Short-Message-Type header indicates the type number of a short message for replacement, see [3GPP TS 23.040].

The header is defined as an extension to the [RFC5322] field definitions. The limits for the occurrence of the field are defined in the following table:

Field	Min Number	Max Number
Replace-Short-Message-Type	0	1

Table 225: Replace-Short-Message-Type header

The field itself is defined in ABNF as follows:

```
replace-short-message-type    = "Replace-Short-Message-Type:"
replace-type-value CRLF
replace-type-value      = "1" / "2" / "3" / "4" / "5" / "6" / "7"
```

Example:

```
Replace-Short-Message-Type: 1
```

C.4. Definition of RCS CPIM Header Extensions

C.4.1. RCS CPIM Extension Name Space

CPIM header extensions make use of the extension framework defined in [RFC3862]. The RCS extensions make use of the RCS name space as defined in this section.

The RCS Namespace is defined as follows:

```
NS: rcs <http://www.gsma.com>
```

NOTE: The namespace is considered as a place holder for a final one to be defined by GSMA or other committees.

C.4.2. Definition of rcs.Service-Centre-Address header

The rcs.Service-Centre-Address header contains the Service Centre Address associated with a short message, see [3GPP TS 23.040]

The header is defined as an extension to the [RFC3862] field definitions. The limits for the occurrence of the field are defined in the following table:

Field	Min Number	Max Number
rcs.Service-Centre-Address	0	1

Table 226: rcs.Service-Centre-Address header

The field itself is defined in ABNF as follows:

```
service-centre-address  = "rcs.Service-Centre-Address:" service-
centre-value CRLF
service-centre-value    = "+" *15DIGIT
```

Example:

```
rcs.Service-Centre-Address: +491712020202
```

C.4.3. Definition of rcs.Reply-Path header

The rcs.Reply-Path header contains the indication whether a reply path exists for a short message, see [3GPP TS 23.040].

The header is defined as an extension to the [RFC3862] field definitions. The limits for the occurrence of the field are defined in the following table:

Field	Min Number	Max Number
rcs.Reply-Path	0	1

Table 227: rcs.Reply-Path header

The field itself is defined in ABNF as follows:

```
reply-path  = "rcs.Reply-Path:" reply-path-value CRLF
reply-path-value  = "0" / "1"
```

Example:

```
rcs.Reply-Path: 0
```

C.4.4. Definition of rcs.Replace-Short-Message-Type header

The rcs.Replace-Short-Message-Type header indicates the type number of a short message for replacement, see [3GPP TS 23.040]

The header is defined as an extension to the [RFC3862] field definitions. The limits for the occurrence of the field are defined in the following table:

Field	Min Number	Max Number
rcs.Replace-Short-Message-Type	0	1

Table 228: Replace-Short-Message-Type header

The field itself is defined in ABNF as follows:

```
rcs-replace-short-message-type = "rcs.Replace-Short-Message-Type:"
                  replace-type-value CRLF

replace-type-value      = "1" / "2" / "3" / "4" / "5" / "6" / "7"
```

Example:

```
rcs.Replace-Short-Message-Type: 1
```

C.4.5. Definition of rcs.Mms-Message-Class header

The rcs.Mms-Message-Class header indicates the class of a multimedia message, see [3GPP TS 23.140].

The header is defined as an extension to the [RFC3862] field definitions. The limits for the occurrence of the field are defined in the following table:

Field	Min Number	Max Number
rcs.Mms-Message-Class	0	1

Table 229: Mms-Message-Class header

The field itself is defined in ABNF as follows:

```
rcs-mms-message-class = "rcs.Mms-Message-Class: "
                  class-value CRLF

class-value = "Personal" / "Advertisement" / "Informational" / "Auto"
```

Example:

```
rcs.Mms-Message-Class: Personal
```

C.4.6. Definition of rcs.Message-Correlator header

The rcs.Message-Correlator header contains the message identification for message correlation, see section 3.2.4.7.2. The encoding of the message identification depends on the context defined in rcs.Message-Context parameter.

The header is defined as an extension to the [RFC3862] field definitions. The limits for the occurrence of the field are defined in the following table:

Field	Min Number	Max Number
rcs.Message-Correlator	0	1

Table 230: Message-Correlator header

The field itself is defined in ABNF as follows:

```
rcs-message-context = "rcs.Message-Correlator: "
                      message-correlator-value CRLF

                ; for encoding rules of "message-correlator-value"
                ; follow references in [RCS-CPM-MSGSTOR-ENDORS]
```

Example:

```
rcs.Message-Correlator: Hello world
```

C.4.7. Definition of rcs.Message-Context header

The rcs.Message-Context header indicates the context and presentation characteristics of a message. It can be used by interpreters to derive the encoding rules for the value of the rcs.Message-Context header.

The header is defined as an extension to the [RFC3862] field definitions. The limits for the occurrence of the field are defined in the following table:

Field	Min Number	Max Number
rcs.Message-Context	0	1

Table 231: Message-Context header

The field itself is defined in ABNF as follows:

```
rcs-message-context = "rcs.Message-Context: "
                      message-context-class CRLF

                ; for encoding rules of " message-context-class"
                ; refer to [RFC3458]
```

Example:

```
rcs.Message-Context: "pager-message"
```

C.5. Extension to Extension ICSI Release Version in User-Agent and Server headers

User-Agent and Server headers are used to indicate the release version and product information of the Extension to Extension Client.

The Extension to Extension Client shall implement the User-Agent and Server headers, according to the rules and procedures of [RFC3261] with the clarifications in this annex.

The User-Agent and Server headers ABNF are specified in [RFC3261] and extended as follows:

```
Server = "Server" HCOLON server-val *(LWS server-val)
User-Agent = "User-Agent" HCOLON server-val *(LWS server-val)
server-val = product / comment
product = ExttoExt-product / token [SLASH product-version]
product-version = token
```

C.5.1. Extension to Extension Version 1.0

This specification allows having several server-val tags. The first of those server-val tags shall be encoding according to the following ABNF:

```
ExttoExt-product = "ExttoExt-" ExttoExt-device-token (SLASH ExttoExt-
product-version)
ExttoExt-device-token = "client" | "serv" token
ExttoExt-product-version = "Ext1.0"
```

Example:

In this example the Extension to Extension Client acting as UAC and the one acting as UAS are Extension to Extension release version 1.0 products. One of the Extension to Extension Clients has inserted its own company and product name and version "ABC-Extensions1000/v1.01".

```
User-Agent: ExttoExt-client/Ext1.0 ABC-Extensions1000/v1.01
Server: ExttoExt-serv/Ext1.0
```

Annex D WebRTC and other ways to access the RCS/IMS network (Informative)

This UNI specification defines a standard way for a device/client to access the RCS functionality in the network to provide end-to-end RCS services. However there are other ways that are different from this UNI specification, that also allow a device/client to access the RCS functionality to provide end to end RCS services. These alternatives are not meant to replace or to modify this UNI specification but to complement it in different deployment environments.

The network architecture and device runtime environment described in this section is to support WebRTC access or UNI interfaces that can be used to expose RCS network functionality to 3rd party clients or operator provided clients (i.e. 1st party clients). Therefore this annex does not mandate any client behaviour for such applications.

NOTE: The text in this Annex is subject to ongoing standardization efforts in 3GPP, OMA, and IETF, and to be updated once the standards become available.

D.1. Introduction to WebRTC

Web Real-Time Communication (RTCWeb/WebRTC) is a suite of IETF and W3C standards (see [IETF-DRAFT-RTCWeb_Overview] and [W3C WebRTC]) that allows web browsers to run real-time media (containing audio, video, and data channels) in a peer-to-peer fashion from web browser to web browser, or between a web browser and a media gateway.

The W3C standards body has defined APIs for browsers to support the media functions of an IP communications client. The browsers support complex media layer handling functions including Codecs, Echo Cancellers, Jitter buffer, NAT traversal and Security functions to make it easier for web developers to integrate real-time communication services into websites and web applications.

WebRTC does not specify an end to end communications network architecture. A network-based architecture for the support of WebRTC client access to IMS is defined in [3GPP TS 23.228]. This specification defines, for example, how WebRTC clients access IMS, reusing IMS client security credentials and/or public identities/credentials as appropriate, the way IMS clients communicate with WebRTC clients connected to IMS and the ability to realise any IMS services to the WebRTC client.

The IETF RTCWeb specifications only partially specify the signalling. In particular, WebRTC requires SDP to be used to describe the media streams involved in the session, and the offer-answer model to negotiate the media but the signalling transport protocol is not specified by IETF.

The specification of the signalling protocol is necessary in order to design a complete WebRTC compatible IMS/RCS end to end architecture. Therefore, in the context of RCS, the following signalling architectures are recommended:

- Signalling using SIP over secure WebSocket as defined by 3GPP [3GPP TS 23.228] and summarised in section D.2.
- Signalling using RESTful NetAPI for WebRTC signalling as defined by OMA [REST WEBRTC SIG API] and summaried in section D.3

D.2. WebRTC RCS Clients using SIP over WebSocket Signalling

The WebRTC IMS architecture in 3GPP [3GPP TS 23.228] enables support of WebRTC clients (see Figure 163).

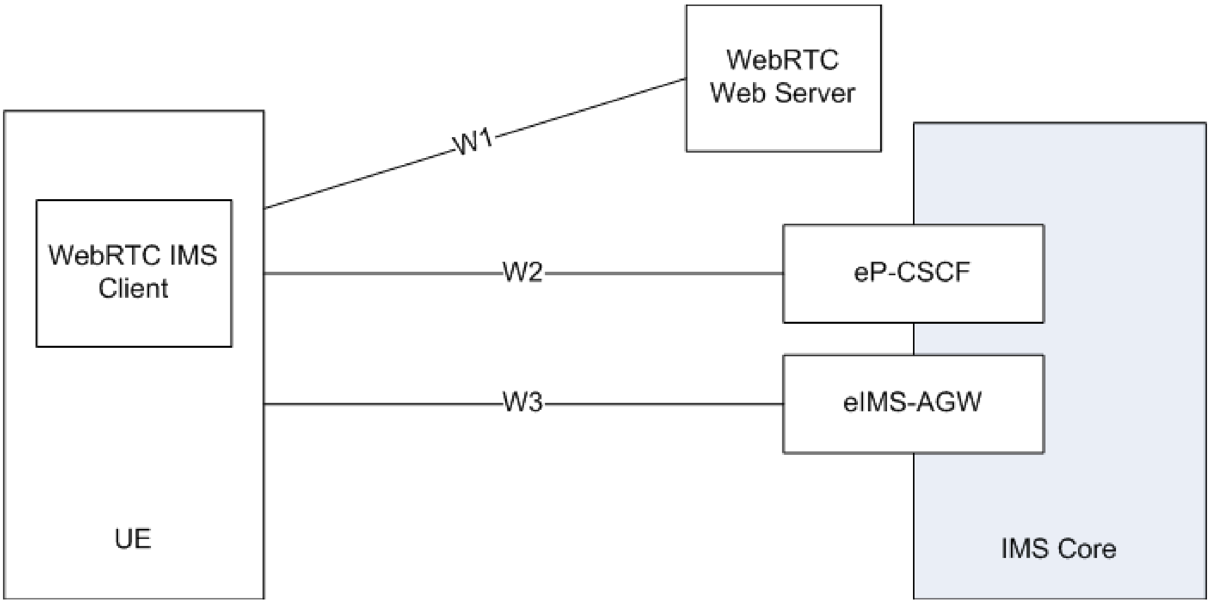

Figure 163: RCS architecture for WebRTC clients – SIP over WebSocket

The figure shows the architecture for RCS clients running in WebRTC enabled browser environments. This architecture supports:

- An interface W1 between the WebRTC IMS Client and the WebRTC Web Server, as described in [3GPP TS 23.228]. HTTPS is normally used to access the web page providing the user interface for the Client and to download the JavaScript application to the browser.
- A signalling plane interface W2 between the WebRTC IMS Client and the enhanced P-CSCF (eP-CSCF) and a Media plane interface W3 between the WebRTC IMS Client and the enhanced IMS-AGW (eIMS-AGW), reusing the UNI interface as specified in this specification but running on protocol stacks adequate for WebRTC clients as described in [3GPP TS 23.228] and summarised in Table 232.

Table 232 summarises the protocols used by WebRTC/SIP RCS clients to access the IMS Core.

Interface	Protocol name	Description	WebRTC RCS Client to gateway Transport layer	WebRTC RCS Client browser API
W2	SIP	Client-IMS core signalling protocol	WebSocket Protocol [RFC6455] SIP over websocket [RFC7118] TCP/IP	WebSocket API as defined in [W3C WS]

W3	MSRP	chat messages, media (pictures) and file exchange protocol	DataChannel transport as defined by [3GPP TS 23.228] U.1.5.1 (i.e. SCTP/DTLS/UDP/IP)	WebRTC API (Data channel control and data access) [W3C WebRTC]
	RTP	Real Time Media (voice and video) exchange	MediaStream track transport As defined by [3GPP TS 23.228] U.1.5.4. (i.e. SRTP/UDP/IP)	WebRTC API (control of real-time media) [W3C WebRTC]

Table 232: RCS protocols for WebRTC clients – SIP over WebSocket

D.3. Device/Clients using RESTful NetAPIs

This architecture option is based on the use of a gateway exposing RCS APIs. It provides following interfaces also summarised in Table 233 and Figure 164:

- An RCSAPI interface will use RESTful Network API for WebRTC Signalling [REST WEBRTC SIG API] and Network API for Notification Channel [REST RCS API] for signalling.
- Interface W3 will only be used for voice and video RTP media.
- The RCSAPI interface also supports other RCS services (e.g. File Transfer, Chat) based on RESTful NetAPI [REST RCS API].

NOTE: The deployment architecture for RCS API Gateway is not specified and open to vendor and operator deployment decision.

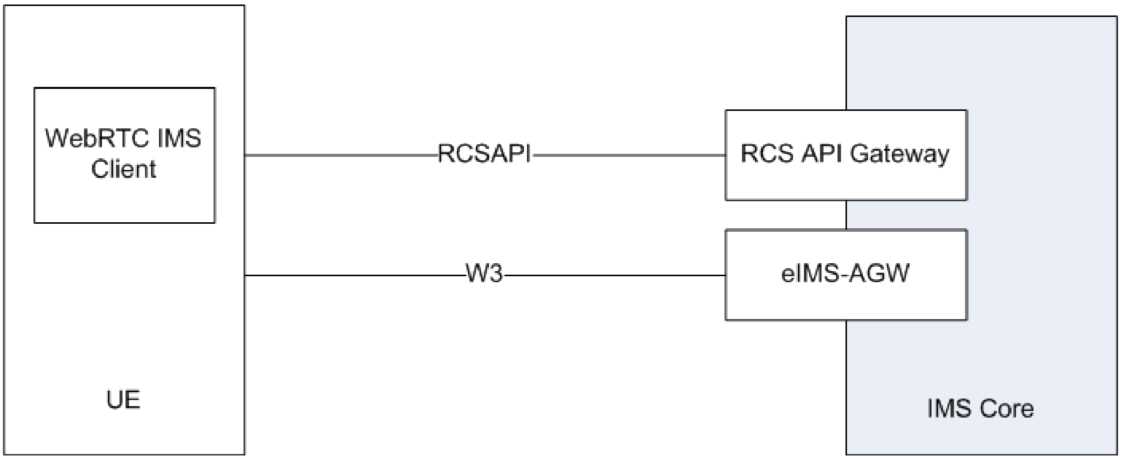

Figure 164: RCS architecture for WebRTC clients – RESTful NETAPI

The following table summarises the list of protocols employed by WebRTC RCS clients using RESTful NETAPIs for signalling and data exchanges.

Interface	Protocol name	Description	WebRTC RCS Client to gateway Transport layer	WebRTC RCS Client browser API
RCS API	RESTful Network API for WebRTC Signalling [REST WEBRTC SIG API]	Client/Gateway-IMS core signalling protocol	HTTP for Network APIs. WebSocket may be used only for notification transport as defined in RESTful Network API Notification Channel.	XMLHttpRequest API as defined in [W3C XHR] for Network APIs. WebSocket API as defined in [W3C WS] may be used only for notification transport as defined in RESTful Network API for Notification Channel.
	Other OMA RESTful NetAPI (e.g. Chat API, File Transfer API) [REST RCS API]	Chat messages, media (pictures) and file exchange protocol	HTTP for NetworkAPIs. WebSocket may be used only for notification transport as defined in RESTful Network API Notification Channel	XMLHttpRequest API as defined in [W3C XHR] for Network APIs. WebSocket API as defined in [W3C WS] may be used only for notification transport as defined in RESTful Network API for Notification Channel.
W3	RTP	Real Time Media (voice and video) exchange	MediaStream track transport As defined by [3GPP TS 23.228] U.1.5.4. (i.e. SRTP/UDP/IP)	WebRTC API (control of real-time media) [W3C WebRTC]

Table 233: RCS protocols for WebRTC clients – RESTful NETAPI

Document Management

Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
1.0	13 August 2012	First version for RCS 5.1 based on approved RCS 5.0 specification version 1.0 Approved by DAG and PSMC	PSMC	Tom Van Pelt / GSMA
1.0	26 September 2012	Added RCC.07 number		Tom Van Pelt / GSMA
2.0	02 May 2013	Applied MCR1001 approved by DAG and PSMC	PSMC	Tom Van Pelt / GSMA
3.0	25 September 2013	Applied MCR1002 approved by DAG and PSMC	PSMC	Tom Van Pelt / GSMA
4.0	28 November 2013	Applied MCR1003 approved by DQR and Global Specification Group (GSG)	GSG	Tom Van Pelt / GSMA
5.0	07 May 2014	First version of the document for RCS 5.2: Include approved CR1004	GSG	Tom Van Pelt / GSMA
6.0	28 February 2015	First version of the document for RCS 5.3: Include approved CR1005	PSMC	Tom Van Pelt / GSMA
7.0	21 March 2016	First version of the document for RCS 6.0: Include approved CR1007	PSMC	Tom Van Pelt / GSMA

Other Information

Type	Description
Document Owner	Network 2020 Programme, Global Specification Group
Editor / Company	Tom Van Pelt / GSMA

It is our intention to provide a quality product for your use. If you find any errors or omissions, please contact us with your comments. You may notify us at prd@gsma.com

Your comments or suggestions & questions are always welcome.