



# Rich Communication Suite 5.3 Endorsement of OMA CPM 2.0 Conversation Functions

Version 4.0

28 February 2015

*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**

Copyright © 2015 GSM Association

## **Disclaimer**

The GSM Association ("Association") makes no representation, warranty or undertaking (express or implied) with respect to and does not accept any responsibility for, and hereby disclaims liability for the accuracy or completeness or timeliness of the information contained in this document. The information contained in this document may be subject to change without prior notice.

## **Antitrust Notice**

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

## Table of Contents

<b>1</b>	<b>Introduction</b>	<b>6</b>
1.1	Overview	6
1.2	Scope	6
1.3	Definition of Terms	6
1.4	Document Cross-References	7
<b>2</b>	<b>References</b>	<b>9</b>
<b>3</b>	<b>Terminology and Conventions</b>	<b>9</b>
<b>4</b>	<b>Introduction</b>	<b>9</b>
4.1	CPM Version 1.0	9
4.2	CPM Version 2.0	10
<b>5</b>	<b>Format of CPM Conversation Items</b>	<b>10</b>
5.1	CPM Standalone Message	10
5.2	CPM Session	10
5.2.1	SDP Contents for CPM Sessions	10
5.3	CPM Conversation Identification	11
5.4	Disposition Notifications	11
5.4.1	Generate Delivery Notification	11
5.4.2	Generate Read Report	11
5.4.3	Receive Delivery Notification	12
5.4.4	Receive Read Report	12
5.4.5	Multidevice handling	13
5.5	“isComposing” Notifications	13
5.6	CPM Service IDs	13
<b>6</b>	<b>Common Procedures</b>	<b>13</b>
6.1	Authenticated Originator’s CPM Address	13
6.1.1	Identifying the sending device in SIP requests and responses	13
6.1.2	Identifying the recipient device in SIP requests and responses	13
6.2	SIP/IP Core	13
6.3	Display Name and Anonymity	14
6.4	Warning Header Field	14
6.4.1	General	14
6.4.2	Warning Texts	14
6.5	Communicating With the ISF	14
6.6	Suitable CPM Clients	14
<b>7</b>	<b>Procedures at CPM Client</b>	<b>14</b>
7.1	Registering at the SIP/IP Core	14
7.2	CPM Standalone Message Handling	15
7.2.1	Sending CPM Standalone Messages	15
7.2.2	Receiving CPM Standalone Messages	16
7.2.3	Deferred CPM Message Handling	16
7.3	CPM Session Handling	17

7.3.1	Initiating New CPM Sessions	17
7.3.2	Receiving a CPM Session Invitation	19
7.3.3	Extending a CPM 1-1 Session to a CPM Group Session	20
7.3.4	Closing a CPM Session	20
7.3.5	Invite other Principals to existing CPM Group Session	20
7.3.6	Remove Participants from a CPM Group Session	20
7.3.7	Modifying a CPM Session	21
7.3.8	Handling a Received CPM Session Modification Request	21
7.3.9	Media Plane Handling for CPM Sessions	21
7.3.10	Participant Information	22
7.3.11	Handling Deferred CPM 1-1 Session Message Delivery	22
7.3.12	Handling Deferred CPM Group Session Message Delivery	22
7.4	CPM File Transfer	22
7.4.1	CPM File Transfer Session Initiation	22
7.4.2	Receiving a CPM File Transfer Request	23
7.4.3	CPM File Transfer Session Release	24
7.4.4	Fetching a Deferred CPM File Transfer file(s)	24
7.4.5	Resuming an interrupted CPM File Transfer	24
<b>8</b>	<b>Procedures at CPM Participating Function</b>	<b>24</b>
8.1	Registration	24
8.1.1	Receive SIP REGISTER Notification	24
8.1.2	Receive Registration Event Information Notifications	24
8.1.3	Terminating the Subscription to Registration Event Information	24
8.1.4	Using the Registration Event Information	24
8.2	Procedures in the Originating Network	25
8.2.1	CPM Standalone Message Handling	25
8.2.2	CPM Session Handling	27
8.2.3	CPM File Transfer Handling	30
8.2.4	Receiving a Disposition Notification	31
8.2.5	Sending a Disposition Notification	31
8.2.6	Media Handling by the Originating Participating Function	31
8.3	Procedures in the Terminating Network	31
8.3.1	CPM Message Handling	31
8.3.2	CPM Session Handling	38
8.3.3	CPM File Transfer Handling	43
8.3.4	Sending a Disposition Notification	44
8.3.5	Receiving a Disposition Notification	44
8.4	User Preferences	44
8.4.1	Retrieving User Preferences	44
8.5	Record CPM Conversation History	44
8.5.1	Record CPM Standalone Message	44
8.5.2	Record CPM Session	45
8.5.3	Record CPM File Transfer	45
<b>9</b>	<b>Procedures at CPM Controlling Function</b>	<b>45</b>

9.1	CPM Standalone Message Handling	45
9.1.1	Pager Mode CPM Standalone Message Handling	45
9.1.2	Large Message Mode CPM Standalone Message Handling	46
9.2	CPM Group Session Handling	47
9.2.1	CPM Group Session Initiation	48
9.2.2	Session Cancellation Request	48
9.2.3	Participant Joining a CPM Group Session for a CPM Pre-defined Group	49
9.2.4	Rejoining CPM Group Session Request	49
9.2.5	Adding Participants Request	50
9.2.6	Removing Participant Request	50
9.2.7	Session Leaving Request	50
9.2.8	CPM Group Session Modification	50
9.2.9	Group Session Ending Request	51
9.2.10	Create Session with a Participant	51
9.2.11	Participant Removing Request	52
9.2.12	Media Plane Handling	52
9.2.13	Pseudonyms in a CPM Group Session	53
9.2.14	Participant Information	53
9.2.15	Disposition Notification	54
9.3	CPM File Transfer Handling	54
9.3.1	CPM File Transfer in Group Session Handling	54
<b>Appendix A.</b>	<b>Change History</b>	<b>54</b>
<b>Appendix B.</b>	<b>Static Conformance Requirements</b>	<b>54</b>
<b>Appendix C.</b>	<b>CPM-Defined Sip Header Fields</b>	<b>54</b>
C.1.	Header Field Definitions	54
C.1.1.	Conversation-ID	54
C.1.2.	Contribution-ID	55
C.1.3.	InReplyTo-Contribution-ID	55
C.1.4.	Session-Replaces	55
C.1.5.	Message-Expires	55
C.1.6.	Message-UID	55
C.1.7.	Contact Header field values	55
C.2.	ABNF for the CPM-defined SIP Headers	55
C.3.	ABNF for the CPM extensions to SIP Headers	55
<b>Appendix D.</b>	<b>Release Version in User-Agent and Server Headers (Normative)</b>	<b>55</b>
D.1.	CPM Version 1.0	55
D.2.	CPM Version 2.0	55
<b>Appendix E.</b>	<b>Examples of CPM-Based Services</b>	<b>55</b>
<b>Appendix F.</b>	<b>The Parameters to be Provisioned for CPM Service</b>	<b>55</b>
<b>Appendix G.</b>	<b>Interoperability With OMA Simple IM Clients</b>	<b>55</b>
G.1.	Media handling with SIMPLE IM Clients	56
G.2.	Handling Message Store	56
<b>Appendix H.</b>	<b>CPM Feature Tags</b>	<b>57</b>

H.1. CPM Feature Identifiers	57
H.2. CPM Client Behaviour	57
H.3. Proposed Formats for CPM Feature Identifiers	57
H.4. Client CPM-based Service Identification	57
H.5. P-Asserted-Service Header Field	57
<b>Appendix I. CPM Notification Formats</b>	<b>57</b>
I.1. Deferred CPM Message Notification Format	57
I.1.1. Out-of-band Deferred CPM Message Notification Format	57
I.1.2. In-band Deferred CPM Message Notification Format	57
I.2. Notification Format of Deferred CPM Message After Expiry	57
<b>Appendix J. Deferred Messages Event Package Definition</b>	<b>57</b>
J.1. Deferred Messages Metadata	57
J.1.1. Structure	57
<b>Appendix K. Format Of Notification For Deferred CPM Message After Expiry</b>	<b>58</b>
<b>Appendix L. Emoticons (Normative)</b>	<b>58</b>
L.1. List of supported emoticons	58
L.2. Emoticon processing by the CPM Clients	58
L.2.1. Emoticons in composed/sent instant messages	58
L.2.2. Emoticons in received/displayed instant messages	58
<b>Document Management</b>	<b>59</b>
Document History	59
Other Information	59

# 1 Introduction

## 1.1 Overview

This document describes which sections of the Open Mobile Alliance Converged IP (Internet Protocol) Messaging (OMA CPM) 2.0 Conversation Functions specification (see [CPMCONVFUNC]) are supported by RCS (Rich Communications Suite) 5.3.

For details on how this fits technically in the RCS 5.3 scope please see [RCS 5.3].

For easier reference, this document follows the same structure as [CPMCONVFUNC]. For that reason the headings of the sections are citations of the headings used in [CPMCONVFUNC], within the sections they describe what part the equivalent section in [CPMCONVFUNC] is supported by RCS. For sections that are not applicable in their entirety, this is mentioned at the top level of the section and the subsections are not mentioned explicitly thereafter. For sections in which no difference with [CPMCONVFUNC] is introduced however, also the subsections are mentioned to state explicitly that they are applicable as well.

This specification lists differences and clarifications for RCS compared to [CPMCONVFUNC]. The former category includes both differences in expected behaviour compared to [CPMCONVFUNC] as well as corrections in behaviour, which should disappear over time when bug fixes will be applied to [CPMCONVFUNC]. The latter category describes what options are chosen for RCS in case [CPMCONVFUNC] provides multiple possibilities and provides clarifications on how the provided functionality is expected to be used.

## 1.2 Scope

This document provides the details of the messaging technology used in this release of RCS.

## 1.3 Definition of Terms

Term	Description
3GPP	3rd Generation Partnership Project
3GPP2	Third Generation Partnership Project 2
ABNF	Augmented Backus–Naur Form
B2BUA	Back to back user agent
CPIM	Common Presence and Instant Messaging
CPM	Converged IP Messaging
GRUU	Globally Routable User Agent URI
HTTP	Hyper-Text Transfer Protocol
IARI	IMS Application Reference Identifier
ICSI	IMS Communication Service Identifier
IETF	Internet Engineering Task Force
IM	Instant Messaging
IMDN	Instant Message Disposition Notification. See [RFC5438].
IMS	IP Multimedia Subsystem
IP	Internet Protocol

ISF	Interworking Selection Function
IWF	Interworking Function
MMS	Multimedia Messaging Service
MSRP	Message Session Relay Protocol
OMA	Open Mobile Alliance
RCS	Rich Communication Suite
RFC	Request For Comments
RTCP	RTP Control Protocol
RTP	Real-time Transport Protocol
SDP	Session Description Protocol
SIMPLE	SIP for Instant Messaging and Presence Leveraging Extensions
SIP	Session Initiation Protocol
SLA	Service Level Agreement
SMS	Short Message Service
SVG	Scalable Vector Graphics
TCP	Transmission Control Protocol
UA	User Agent
UAC	User Agent Client
UAS	User Agent Server
URI	Uniform Resource Identifier
XDM	XML Document Management
XDMS	XML Document Management Server
XML	eXtensible Markup Language

#### 1.4 Document Cross-References

Ref	Document Number	Title
1	[RCS 5.3]	GSMA PRD RCC.07 - RCS 5.3 - Advanced Communications: Services and Client Specification Version 6.0, 28 February 2015 <a href="http://www.gsma.com/rcs/">http://www.gsma.com/rcs/</a>
2	[RCS5-IMENDORS]	GSMA PRD RCC.12 - RCS 5.3 Endorsement of OMA SIP/SIMPLE IM 2.0, Version 4.0, 28 February 2015 <a href="http://www.gsma.com/rcs/">http://www.gsma.com/rcs/</a>
3	[CPMCONVFUNC]	CPM Conversation Functions, Open Mobile Alliance Ltd. OMA-TS-CPM_Conv_Funct-V2_0-20150113-C <a href="http://member.openmobilealliance.org/ftp/Public_documents/COM/COM-CPM/Permanent_documents/OMA-TS-CPM_Conversation_Function-V2_0-20150113-C.zip">http://member.openmobilealliance.org/ftp/Public_documents/COM/COM-CPM/Permanent_documents/OMA-TS-CPM_Conversation_Function-V2_0-20150113-C.zip</a>

4	[SIMPLEIM]	Instant Messaging using SIMPLE, Open Mobile Alliance Ltd. OMA-TS-SIMPLE_IM-V2_0-20130809-D <a href="http://member.openmobilealliance.org/ftp/Public_documents/COM/IM/Permanent_documents/OMA-TS-SIMPLE_IM-V2_0-20130809-D.zip">http://member.openmobilealliance.org/ftp/Public_documents/COM/IM/Permanent_documents/OMA-TS-SIMPLE_IM-V2_0-20130809-D.zip</a>
5	[MMS1.3Conf]	MMS Conformance Document, Open Mobile Alliance Ltd. OMA-TS-MMS-CONF-V1_3-20110913-A <a href="http://www.openmobilealliance.org/Technical/release_program/docs/MMS/V1_3-20080128-C/OMA-TS-MMS-CONF-V1_3-20110913-A.pdf">http://www.openmobilealliance.org/Technical/release_program/docs/MMS/V1_3-20080128-C/OMA-TS-MMS-CONF-V1_3-20110913-A.pdf</a>
6	[TS26141]	3GPP TS 26.141, IP Multimedia System (IMS) Messaging and Presence; Media formats and codecs; V9.0.0, December 2009 <a href="http://www.3gpp.org">http://www.3gpp.org</a>
7	[RFC3265]	Session Initiation Protocol (SIP)-Specific Event Notification, IETF, June 2002 <a href="http://www.ietf.org/rfc/rfc3265.txt?number=3265">http://www.ietf.org/rfc/rfc3265.txt?number=3265</a>
8	[RFC3994]	Indication of Message Composition for Instant Messaging, IETF, January 2005 <a href="http://www.ietf.org/rfc/rfc3994.txt?number=3994">http://www.ietf.org/rfc/rfc3994.txt?number=3994</a>
9	[RFC4028]	Session Timers in the Session Initiation Protocol (SIP), IETF, April 2005 <a href="http://www.ietf.org/rfc/rfc4028.txt">http://www.ietf.org/rfc/rfc4028.txt</a>
10	[RFC4145]	TCP-Based Media Transport in the Session Description Protocol (SDP), IETF, September 2005 <a href="http://www.ietf.org/rfc/rfc4145.txt?number=4145">http://www.ietf.org/rfc/rfc4145.txt?number=4145</a>
11	[RFC4575]	A Session Initiation Protocol (SIP) Event Package for Conference State, IETF, August 2006 <a href="http://www.ietf.org/rfc/rfc4575.txt?number=4575">http://www.ietf.org/rfc/rfc4575.txt?number=4575</a>
12	[RFC4975]	The Message Session Relay Protocol (MSRP), IETF, September 2007 <a href="http://www.ietf.org/rfc/rfc4975.txt?number=4975">http://www.ietf.org/rfc/rfc4975.txt?number=4975</a>
13	[RFC5364]	Extensible Markup Language (XML) Format Extension for Representing Copy Control Attributes in Resource Lists, IETF, October 2008 <a href="http://www.ietf.org/rfc/rfc5364.txt">http://www.ietf.org/rfc/rfc5364.txt</a>
14	[RFC5366]	Conference Establishment using Request-contained lists in the Session Initiation Protocol (SIP), IETF, October 2008 <a href="http://www.ietf.org/rfc/rfc5366.txt?number=5366">http://www.ietf.org/rfc/rfc5366.txt?number=5366</a>
15	[RFC5547]	A Session Description Protocol (SDP) Offer/Answer Mechanism to Enable File Transfer, IETF, May 2009 <a href="http://www.ietf.org/rfc/rfc5547.txt">http://www.ietf.org/rfc/rfc5547.txt</a>
16	[RFC6135]	Alternative Connection Model for the Message Session Relay Protocol (MSRP) IETF RFC <a href="http://tools.ietf.org/html/rfc6135">http://tools.ietf.org/html/rfc6135</a>



17	[RFC6172]	Connection Establishment for Media Anchoring (CEMA) for the Message Session Relay Protocol (MSRP) IETF RFC <a href="http://tools.ietf.org/html/rfc6172">http://tools.ietf.org/html/rfc6172</a>
18	[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) <a href="http://www.3gpp.org">http://www.3gpp.org</a>

## 2 References

See chapter 1.4.

## 3 Terminology and Conventions

The same conventions, terminology, definitions and abbreviations used in chapter 3 of [CPMCONVFUNC] are valid for RCS. Additional abbreviations and terms specific for this document are in chapter 1.3.

## 4 Introduction

Note: RCS supports the following modes of CPM-based communication

- One-to-One and One-to-Many Pager Mode Standalone messaging including interworking
- One-to-One and One-to-Many Large Message Mode Standalone Messaging including interworking
- Deferred Messaging
- File Transfer
- One-to-one session and Ad-hoc session mode messaging including interworking

RCS does not support the following modes of CPM-based communication

- Pre-defined group messaging

NOTE: Support for File Transfer, One-to-One and Ad-Hoc group sessions is in RCS also provided based on SIMPLE (SIP for Instant Messaging and Presence Leveraging Extensions) IM (Instant Messaging, see [RCS5-IMENDORS]).

### 4.1 CPM Version 1.0

Following differences with [CPMCONVFUNC]:

- In CPM group session handling the case for pre-defined groups is not applicable for RCS
- In CPM group session handling the case for removal of participants is not applicable for RCS
- The support of continuous media types is not applicable for RCS

- User Preference Profiles are not applicable for RCS, as all users will make use of the same predefined preferences. An individual user does not have the possibility to change his preferences nor to have preferences specific for particular devices.
- Recording of messages, sessions and file transfers is not subject to the CPM/RCS user's preferences, but all messages sessions and file transfers are temporarily recorded for device synchronisation purposes. A user can decide after that to store the message, session or file transfer permanently.

## 4.2 CPM Version 2.0

No differences with [CPMCONVFUNC].

# 5 Format of CPM Conversation Items

## 5.1 CPM Standalone Message

No differences with [CPMCONVFUNC].

## 5.2 CPM Session

Following difference with [CPMCONVFUNC]:

- The case for a pre-defined group (including the termination of the session) is not applicable for RCS

### 5.2.1 SDP Contents for CPM Sessions

#### 5.2.1.1 SDP Contents when Initiating or Modifying a CPM Session

Following differences with [CPMCONVFUNC]:

- The modification of a CPM session (including the use of a CPM session modification request and the procedures for modified Session Description Protocol (SDP)) is not applicable to RCS
- The use of real-time continuous media, using RTP (Real Time Protocol) / Real Time Control Protocol) RTCP is not applicable for RCS (in combination with CPM that is) nor is the use of any other Media Stream type besides MSRP (Message Session Relay Protocol).
- The reference to [GSMA IR.92] is not applicable for RCS (in combination with CPM that is)

As a clarification for RCS:

- As described in [RCS 5.3], an RCS client shall always support receiving "isComposing" notifications and thus include the value *"application/im-iscomposing+xml"* in the *a=accept-types* attribute
- As described in [RCS 5.3], an RCS client shall always support receiving disposition notifications and thus include the value *"message/imdn+xml"* in the *a=accept-wrapped-types* attribute

#### 5.2.1.2 SDP Handling at Intermediate Nodes

Following difference with [CPMCONVFUNC]:

- The use of RCP/RTCP based media (including the reference to [RFC3550]) is not applicable for RCS

### 5.2.1.3 SDP Handling at Terminating Nodes

Following difference with [CPMCONVFUNC]:

- The use of real-time continuous Media, using RTP/RTCP is not applicable for RCS (in combination with CPM that is) nor is the use of any other media stream type besides MSRP.

### 5.2.1.4 Handling of Media connection parameters for MSRP

No differences with [CPMCONVFUNC].

#### 5.2.1.4.1 Legacy MSRP session matching

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- This is used by the Participating Function with legacy CPM clients (e.g. CPM v1.0 compliant).

## 5.3 CPM Conversation Identification

No differences with [CPMCONVFUNC].

## 5.4 Disposition Notifications

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- Local Device settings shall not be applicable to the sending of delivery notifications.

### 5.4.1 Generate Delivery Notification

Following differences with [CPMCONVFUNC]:

- The CPM Feature Tag populated in the Accept-Contact header field, e.g. urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session for any Delivery Notification pertaining to a Chat message, will be 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 for the Accept-Contact header field) as indicated in section 7 "Procedures at CPM Client" of [CPMCONVFUNC].
- Step 2, b, 2 is applicable also when the disposition notification is sent using a SIP MESSAGE request
- Step 2.b. – support for the NOTE is mandatory for RCS.

As a clarification for RCS:

- The IMDN shall be sent back in the MSRP method only if the chat session is established with the original device that sent that message.

### 5.4.2 Generate Read Report

Following differences with [CPMCONVFUNC]:

- The CPM Feature Tag populated in the Accept-Contact header field of the IMDN display notification will be 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 for the Accept-Contact header field) as indicated in section 7 “Procedures at CPM Client” of [CPMCONVFUNC]
- In step 2 when sending the disposition notification to a CPM Chat Message through a SIP MESSAGE in step 1 as specified in [RFC5438] any address received in further IMDN-Record-Route headers shall be included in additional IMDN-Route headers.
- In step 2 when sending the disposition notification to a CPM Chat Message through a SIP MESSAGE in step 2, b the Request URI shall be set to the authenticated originator’s CPM Address of the received SIP INVITE request if no GRUU was included in the Contact Header field.
- In step 2, when sending the disposition notification to a CPM Chat Message, in step 4 the CPIM TO URI shall be set to the authenticated originator’s CPM Address of the received SIP INVITE request if a GRUU was not provided in the Contact Header Field of the received SIP INVITE request.
- In step 2, when sending the disposition notification to a CPM Chat Message through a SIP MESSAGE in step 5 the CPM Client shall include in the Conversation-ID header field the Conversation-ID included in the received SIP INVITE request.
- Step 2, step 2 when sending the disposition notification through the MSRP procedure is applicable also when the disposition notification is sent using a SIP MESSAGE request
- In step 2, in step 1, ii when sending the disposition notification through the MSRP procedure in a Group Chat, the CPIM From header may depending on local service provider policy (see section 2.5 of [RCS 5.3]) also be set to the tel URI of the CPM User. In a 1-to-1 Chat it shall be set to *sip:anonymous@anonymous.invalid*.
- In step 2, in step 2 1, iii when sending the disposition notification through the MSRP procedure in a Group Chat, the CPIM To header may depending on local service provider policy (see section 2.5 of [RCS 5.3]) also be set to the tel URI of the CPM User. In a 1-to-1 Chat it shall be set to *sip:anonymous@anonymous.invalid*.

As a clarification for RCS:

- The IMDN shall be sent back in the MSRP method only if the chat session is established with the original device that sent that message.

### 5.4.3 Receive Delivery Notification

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- For RCS, a CPM client may receive disposition notifications from a legacy client (e.g. SIMPLE IM) with a different Conversation-ID and/or Contribution-ID than the 1-to-1 Chat in which the message to which the notification applies was sent. An RCS client should therefore match received notifications to the messages sent based only on the IMDN Message-ID.

### 5.4.4 Receive Read Report

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- For RCS, a CPM client may receive disposition notifications from a legacy client (e.g. SIMPLE IM) with a different Conversation-ID and/or Contribution-ID than the 1-to-1 Chat in which the message to which the notification applies was sent. An RCS client should therefore match received notifications to the messages sent based only on the IMDN Message-ID.

#### **5.4.5 Multidevice handling**

No differences with [CPMCONVFUNC].

### **5.5 “isComposing” Notifications**

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- Also in a Group Chat disposition notifications shall not be requested when an RCS client sends “isComposing” notifications
- “isComposing” notifications shall not be stored

### **5.6 CPM Service IDs**

No differences with [CPMCONVFUNC].

## **6 Common Procedures**

### **6.1 Authenticated Originator’s CPM Address**

Following differences with [CPMCONVFUNC]:

- Anonymity is not applicable for RCS
- Pre-defined group addresses are not applicable for RCS

#### **6.1.1 Identifying the sending device in SIP requests and responses**

Following differences with [CPMCONVFUNC]:

- If no SIP URI is present then the TEL URI shall be used, following provisioning procedure in section 2.5 of [RCS 5.3].

As a clarification for RCS:

- The value used for the instance identifier shall be determined as specified in section 2.4.2 of [RCS 5.3].

#### **6.1.2 Identifying the recipient device in SIP requests and responses**

No differences with [CPMCONVFUNC].

### **6.2 SIP/IP Core**

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- The SIP/ Internet Protocol (IP) core shall always correspond to Third Generation Partnership Project (3GPP/3GPP2) (IP Multimedia Subsystem) IMS

### 6.3 Display Name and Anonymity

Following differences with [CPMCONVFUNC]:

- Anonymity is not supported, therefore a RCS Client shall never include a Privacy header containing privacy type “id” nor act upon such header
- An RCS client shall include the display name in both the P-Preferred-Identity and the From header
- Predefined Groups are not supported for RCS, therefore the first priority case for the controlling function to determine the display name is not applicable for RCS

### 6.4 Warning Header Field

#### 6.4.1 General

No differences with [CPMCONVFUNC].

#### 6.4.2 Warning Texts

No differences with [CPMCONVFUNC].

### 6.5 Communicating With the ISF

Following differences with [CPMCONVFUNC]:

- The case for sending a CPM File Transfer Invitation to the ISF (Interworking Selection Function) is not applicable for RCS

As a clarification for RCS:

- The CPM Participating or CPM Controlling Function communicating with the ISF (Interworking Selection Function) will always act as a B2BUA (Back to Back User Agent). For MESSAGE requests, it is an implementation decision whether to act as a B2BUA or as a proxy.

### 6.6 Suitable CPM Clients

No differences with [CPMCONVFUNC].

## 7 Procedures at CPM Client

No differences with [CPMCONVFUNC].

### 7.1 Registering at the SIP/IP Core

Registration will be done as described in [RCS 5.3]. This means the following differences with [CPMCONVFUNC]:

- In step 4, the feature tags, ICSIs and IARIs (IMS Application Reference Identifier) for the other RCS use cases and enablers will be included.

As a clarification for RCS:

- In Step 3 and Note 1: The value used for the instance identifier shall be determined as specified in section 2.4.2 of [RCS 5.3].
- For step 6 an RCS client shall include a Supported header field with the option tag ‘gruu’.

- RCS devices may need to pull Deferred CPM Standalone Messages and will thus subscribe to the “deferred-messages” event package.
- As described in [RCS 5.3], the network is not required to support the use of GRUUs. In that case the client will not receive any GRUUs in the response to the REGISTER request. The client shall then include the sip.instance parameter and value used in step 3 in the Contact header of non-REGISTER requests and responses.

## 7.2 CPM Standalone Message Handling

No differences with [CPMCONVFUNC].

### 7.2.1 Sending CPM Standalone Messages

#### 7.2.1.1 Sending a Pager Mode CPM Standalone Message

Following differences with [CPMCONVFUNC]:

- Anonymity is not supported: Step 6 is not applicable for RCS
- Pre-defined groups are not supported: Step 10 is not applicable for RCS

As clarifications for RCS:

- Multiple authenticated originators' CPM addresses may be received (see [RCS 5.3]). The address used when sending a reply shall be based on the provisioning setting described in [RCS 5.3].
- In step 9 c and step 9 d, if the group message is a reply to a previously received group message, also the addresses of the resource-list received in that previously received message will initially be added to the resource-list with the same “To” and “CC” qualifiers that were used in that previous received list. The user will have the option though to remove addresses from the list or to include additional recipients. This will provide a “reply-all” kind of functionality.

#### 7.2.1.2 Sending a Large Message Mode CPM Standalone Message

Following differences with [CPMCONVFUNC]:

- Anonymity is not supported: Step 7 is not applicable for RCS
- Pre-defined groups are not supported: Step 11 is not applicable for RCS
- If an error response is received on the INVITE request, an ACK request shall be sent to acknowledge its reception
- If an error response is received on one of the MSRP SEND requests or the TCP (Transmission Control Protocol) connection used for the MSRP session is lost, no further requests shall be sent and the session shall be terminated by following the same procedure as when it had been successfully transferred.
- Once a response has been received to the BYE request that was sent the resources on the media plane shall be released

As clarifications for RCS:

- If a response different from 200 “OK” is received on the INVITE or one of the MSRP SEND requests, the user shall be informed that the message could not be delivered
- Multiple authenticated originators' CPM addresses may be received (see [RCS 5.3]). The address used when sending a reply shall be based on the provisioning setting described in [RCS 5.3].

- In step 9 c and step 9 d, if the group message is a reply to a previously received group message, also the addresses of the resource-list received in that previously received message will initially be added to the resource-list with the same “To” and “CC” qualifiers that were used in that previous received list. The user will have the option though to remove addresses from the list or to include additional recipients. This will provide a “reply-all” kind of functionality.

### **7.2.1.3 Generate a CPM Standalone Message**

Following differences with [CPMCONVFUNC]:

- Step 2 c (the sending of CPM Standalone Message to a pre-defined group) is not applicable for RCS

### **7.2.1.4 Forwarding/Including Stored Data without Downloading to the CPM Client**

Not applicable for RCS

## **7.2.2 Receiving CPM Standalone Messages**

### **7.2.2.1 Receiving a Pager Mode CPM Standalone Message and SIP IMDNs**

No differences with [CPMCONVFUNC].

### **7.2.2.2 Receiving a Large Message Mode CPM Standalone Message**

Following differences with [CPMCONVFUNC]:

- In step 1, also the types included in the accept-wrapped-types attribute will be checked
- If the session is terminated before the complete standalone message was relayed or the TCP connection used for the MSRP session is lost before the transmission was completed, the received contents will be silently discarded
- After sending a response to a BYE request an RCS client will release the media plane resources

### **7.2.2.3 Handling of Received CPM Standalone Messages and SIP IMDNs**

Following differences with [CPMCONVFUNC]:

- Step 2 (references to an external body) is not applicable for RCS

## **7.2.3 Deferred CPM Message Handling**

### **7.2.3.1 Subscribe to Deferred CPM Message Info**

Following difference with [CPMCONVFUNC]:

- In step 2: the CPMDDeferredMsgMgmtURI address will be provisioned through the DEFERRED-MSG-FUNC-URI parameter described in [RCS 5.3].

As a clarification for RCS:

- A client will send a SUBSCRIBE request as described in this section shortly after its initial registration. For as long as it remains registered, it will send no further SUBSCRIBE requests.



- For RCS, depending on device settings and situation (for instance whether or not it is roaming), the client can also immediately fetch all deferred messages rather than presenting the list to the user

### **7.2.3.2 Handling Deferred CPM Message(s)**

#### **7.2.3.2.1 Handling Deferred CPM Message(s) before Having Received an Expiry Notification**

Following differences with [CPMCONVFUNC]:

- Step 6 b is not applicable for RCS
- Step 7 a iii and 7 a iv are not applicable for RCS.

As a clarification for RCS:

- In step 2: the CPMDeferredMsgMgmtURI will be provisioned as described for step 2 of section 7.2.3.1
- An RCS client shall handle Deferred CPM Chat Messages as described in [RCS 5.3].
- For NOTE 1: depending on Device settings and circumstances, the CPM client may also initiate the procedure automatically in case it knows that deferred messages are available

#### **7.2.3.2.2 Retrieving CPM Standalone Message(s) after Receiving a Notification for Expiry of Deferred CPM Message(s)**

Not applicable for RCS

#### **7.2.3.3 Receiving a notification for Expiry of Deferred CPM Standalone Message**

Not applicable for RCS

#### **7.2.3.4 Notification Handling**

##### **7.2.3.4.1 Out-of-band Notifications**

Not applicable for RCS

##### **7.2.3.4.2 In-band Notifications**

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- depending on Device settings and circumstances, the CPM client may also initiate the procedure to fetch all deferred messages automatically

### **7.3 CPM Session Handling**

#### **7.3.1 Initiating New CPM Sessions**

##### **7.3.1.1 Initiating a CPM 1-1 Session**

Following difference with [CPMCONVFUNC]:

- Anonymity is not supported: For RCS in Step 7 if the network supports GRUU (see section 7.1) it shall always be the public GRUU which is included. On networks where

GRUU is not supported a sip.instance parameter with a value as defined in [RCS 5.3] will be included instead.

- As anonymity is not supported step 8 is not applicable for RCS.
- When a response is received to the INVITE request that differs from 200 OK, the CPM client will send an ACK request to acknowledge its reception
- Step 5 shall be done as described in the addressing section in [RCS 5.3].

As a clarifications for RCS:

- If a new INVITE request is sent to the same user before an outstanding one has been answered (see [RCS 5.3]), both INVITE requests shall use different values for the Contribution-ID.
- For step 9, an RCS client shall include the Supported header field
- When a 200 OK response is received on an INVITE request from a user with whom a one-to-one CPM session is established already, an RCS client will terminate the existing session by sending a SIP BYE request. This situation may happen when multiple INVITE requests has been sent to a user and more than one of these request are answered with a 200 OK.
- When a response is received to the INVITE request that differs from 200 OK or 486 Busy Here, the user will be informed that the session could not be set-up  
Note as described in chapter 7.3.2 of this document re-direction is not supported in RCS.

#### **7.3.1.2 Initiating a CPM Group Session for a CPM Ad-hoc Group**

Following difference with [CPMCONVFUNC]:

- Anonymity is not supported: step 7 is not applicable for RCS.
- For RCS in step 9 the CPM Client shall set the Contribution-ID to the same value as the Conversation-ID since in RCS there is only one Group Chat within a conversation.
- When a response is received to the INVITE request that differs from 200 OK, the CPM client will send an ACK request to acknowledge it's reception

As a clarification for usage of [RFC5366] in step 5:

- Only the mandatory parts of the URI-List are applicable for RCS
- RCS clients will not use the possibility to indicate that some contacts would be invited as "CC" or "BCC", nor will they indicate that some contacts have to be anonymized.

As a clarification for RCS:

- In step 3, when supporting File Transfer in a Group Chat, File Transfer via HTTP (Hyper-Text Transfer Protocol) and/or Geolocation PUSH the client shall take into account the differences and clarifications provided in [RCS 5.3].
- When a response is received to the INVITE request that differs from 200 OK, the user will be informed that the session could not be set-up

#### **7.3.1.3 Initiating a CPM Group Session for a CPM pre-defined Group**

Not applicable for RCS

#### **7.3.1.4 Joining a CPM Group Session for a Join-in Group**

Not applicable for RCS

### 7.3.1.5 Re-joining a CPM Long-lived Group Session

Following difference with [CPMCONVFUNC]:

- As anonymity is not supported step 7 including the NOTE is not applicable for RCS.
- When a response is received to the INVITE request that differs from 200 OK, the CPM client will send an ACK request to acknowledge its reception
- Step 5 shall be done as described in the addressing section in [RCS 5.3].

As a clarifications for RCS:

- An RCS client shall include the public GRUU or +sip.instance tag with the appropriate value in the Contact header field as described in section 6.1.1
- If a new INVITE request is sent to the same user before an outstanding one has been answered (see [RCS 5.3]), both INVITE requests shall use different values for the Contribution-ID.
- For step 9, an RCS client shall include the Supported header field
- When a 200 OK response is received on an INVITE request from a user with whom a one-to-one CPM session is established already, an RCS client will terminate the existing session by sending a SIP BYE request. This situation may happen when multiple INVITE requests has been sent to a user and more than one of these request are answered with a 200 OK.
- When a response is received to the INVITE request that differs from 200 OK, the user will be informed that the session could not be set-up

NOTE: as described in chapter 7.3.2 of this document re-direction is not supported in RCS.

### 7.3.2 Receiving a CPM Session Invitation

Following difference with [CPMCONVFUNC]:

- In step 1 a, the case for temporarily declining a session is not applicable for RCS
- Steps 1 b and 1 c are not applicable for RCS

Following clarifications are given for RCS:

- Step 1 d is applicable for RCS only for the case of a Group Chat
- The invited RCS device shall respond with SIP 180 to announce to inviting user that the CPM session invite has reached the invited user.
- If a participant list according to [RFC5366] is present in the invitation, the client shall communicate it to the user.
- In step 3 c, an RCS client shall correlate the CPM Address of the inviting user with the client's address book in order to derive a display name for the user
- In step 3 c: INVITE requests from other RCS users will not include a Privacy header with the value set to 'id'. Such requests may occur due to interworking with non-RCS systems though.
- In step 6, when supporting File Transfer in a Group Chat the client shall take into account the differences and clarifications provided in [RCS 5.3].

### 7.3.3 Extending a CPM 1-1 Session to a CPM Group Session

Following difference with [CPMCONVFUNC]:

- Anonymity is not supported: For RCS in step 5 it shall always be the public GRUU or +sip.instance parameter which is included
- As anonymity is not supported step 7 is not applicable for RCS

As a clarifications for RCS:

- In step 3, when supporting File Transfer in a Group Chat the client shall take into account the differences and clarifications provided in [RCS 5.3].

### 7.3.4 Closing a CPM Session

#### 7.3.4.1 Closing a CPM 1-1 Session

No differences with [CPMCONVFUNC].

#### 7.3.4.2 Leaving a CPM Group Session

Only one difference with [CPMCONVFUNC]: the section on anonymity, step 3, is not applicable for RCS.

#### 7.3.4.3 Receiving a CPM Session Closing Request

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- For RCS no specific behaviour will be provided in case a Reason Header is included with the protocol set to "SIP" and the protocol-cause set to "200".

#### 7.3.4.4 Receiving a CPM Session Cancellation

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- For RCS as described in section 2.11 of [RCS 5.3], when a Reason Header is included with the protocol set to "SIP" and the protocol-cause set to "200", a group chat or File Transfer session for which an explicit invitation was shown to the user as a result of the configuration described in [RCS 5.3] will not be indicated to the user appropriately and therefore not as a "missed call" or similar.

### 7.3.5 Invite other Principals to existing CPM Group Session

Only one differences with [CPMCONVFUNC]:

- the section on anonymity (step7) including the NOTE is not applicable for RCS.

As a clarification for RCS:

- An RCS client shall be aware that a CPM Group Session is a Closed CPM Group Session.

### 7.3.6 Remove Participants from a CPM Group Session

Not applicable for RCS: this is seen as a quite advanced use case.

### 7.3.7 Modifying a CPM Session

Not applicable for RCS

### 7.3.8 Handling a Received CPM Session Modification Request

An RCS client receiving a SIP re-INVITE request that includes a new SDP offer in the body SHALL reject the request with a SIP 488 "Not Acceptable Here" response according to the rules and procedures of SIP/IP Core and exit this procedure.

### 7.3.9 Media Plane Handling for CPM Sessions

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- An RCS client will support the emoticons defined in [Appendix L](#)
- An RCS client will support all static and continuous media formats described in [TS26141] apart from vector graphics. As the support for the entire Scalable Vector Graphics (SVG) Tiny is optional in [TS26141] and not required for any other RCS service, the support for vector graphics in CPM sessions is optional depending on client capabilities.
- Next to plain text as described in [TS26141], an RCS client will also support Rich Text as defined in chapter 7.1.9.2.1 of [MMS1.3Conf]

#### 7.3.9.1 MSRP-based Media Streams

Following difference with [CPMCONVFUNC]:

- Step 1 a is applicable for RCS only for Delivery and Display notifications in group sessions (i.e. there is no private messaging in a Group Chat for RCS). In other cases the CPIM To header will be set to *sip:anonymous@anonymous.invalid*.
- The section on pre-defined groups (step 1 c) is not applicable for RCS.
- RCS clients will send isComposing Messages as described in [CPMCONVFUNC] chapter 5.5.

Following clarifications are given for RCS:

- In 1-to-1 sessions, the CPIM From header will be set to *sip:anonymous@anonymous.invalid*.
- In ad-hoc group sessions, the CPIM From header will be set to the sending user's identity including the display name as Formal Name.
- An RCS client can include a Content-Disposition header with the value set to Attachment in the MSRP SEND request in order to perform an in-session file transfer.
- All RCS clients shall send text with *text/plain* content type inside the cpim wrapper.
- An RCS client shall offer the user to send the content using the File Transfer mechanism if the content to be sent in a 1-to-1 chat is larger than the value of the parameter for "maximum allowed content size in 1-to-1 chat" defined in [RCS 5.3]
- An RCS client shall notify the user that the content is not possible to send in a group chat if the content to be sent is larger than the value of the parameter for "maximum allowed content size in group chat" defined in [RCS 5.3].

Alternatively, the RCS client may offer the possibility to adapt the content to fit within the configured limit after consulting the user.

### 7.3.9.2 RTP/RTCP-based Media Streams

Not applicable for RCS: RTP/RTCP is not used within the context of CPM

## 7.3.10 Participant Information

### 7.3.10.1 Subscribe to Receiving CPM Group Session Participant Information

No differences with [CPMCONVFUNC].

### 7.3.10.2 Receive Participant Information Notification

No difference with [CPMCONVFUNC].

Following clarifications are given for RCS:

- For RCS the chat Blacklist URI shall be as defined in section 2.14 of [RCS 5.3].

### 7.3.11 Handling Deferred CPM 1-1 Session Message Delivery

Following difference with [CPMCONVFUNC]:

- In step 4 for RCS the case of a CPM client responding with a 480 Temporarily not available error response is not applicable for RCS

As a clarification for RCS:

- In step 2, the session is set up as sendonly and will be identified by the user part of the URI in the P-Asserted-Identity header being rcse-standfw. This session shall only be used to deliver stored IMDNs and an RCS client shall automatically accept this session

### 7.3.12 Handling Deferred CPM Group Session Message Delivery

Following difference with [CPMCONVFUNC]:

- As described in [RCS 5.3], an RCS client that lost connectivity shall automatically attempt to rejoin when regaining connectivity. The CPM recommendation for the client to only rejoin when there are user messages to send is therefore not applicable for RCS.

## 7.4 CPM File Transfer

RCS has the following restrictions on file transfer:

- One-to-Many File Transfer is only supported in Group Chat
- Only one file is sent per file transfer session.
- Only sending of files is supported for RCS. Requesting files is not part of the RCS use cases

### 7.4.1 CPM File Transfer Session Initiation

Following differences with [CPMCONVFUNC]:

- In step 1 the procedure for initiating a file transfer to recipient pre-defined group is not relevant for RCS. That means that the reference to chapter 7.3.1.3 is not applicable for RCS
- Only one file can be sent in a file transfer session. Step 1 e, 2 a and 2 b are thus not applicable for RCS

- MSRP Success Reports are not requested: The reference to it in step 2 is thus not applicable for RCS.

As clarifications for RCS:

- Step 1, the case for sending a file to multiple recipients is only applicable in combination with a group chat. Furthermore in that case the generated INVITE request shall comply with the differences and clarifications on File Transfer in Group Chat provided in section 3.5.4.2 of [RCS 5.3] and section 7.4.1.1.
- The case for pulling a file described in step 1. d. ii. is for RCS only applicable in case of resume (see section 7.4.5) or timeout of the request as described in section 7.4.4
- For step 1. g. for RCS the capacity of the recipient to support a thumbnail shall be determined before initiating a File Transfer request through the RCS capability exchange as described in sections 2.6 and 2.7 of [RCS 5.3]
- For step 1. h. for RCS only delivery notifications shall be requested for File Transfer.
- Usage of the SDP attributes provided in [RFC5547] shall be identical to what is described in [IR.79] chapter 3.4. The only exception is the use of the file-disposition attribute. For File Transfer this will be set to 'attachment' or 'render' depending on the requested service. For a file transfer, 'attachment' shall be used. During an Image Sharing session, 'render' is used to indicate an immediate display of the image.
- For RCS File transfer, message/cpim wrapped MSRP requests shall not be used.
- If the file to be transferred is larger than the value of the parameter for "maximum allowed file size for file transfer" defined in [RCS 5.3], the RCS client shall notify the user that the file is not possible to send. Alternatively, the RCS client may offer the possibility to adapt the content to fit within the configured limit after consulting the user.
- As disposition notifications only positive-delivery notifications will be requested.

#### 7.4.1.1 CPM File Transfer Session Initiation in CPM Session

No differences with [CPMCONVFUNC].

As clarifications for RCS:

- For RCS, the described procedure is only applicable for a transfer in a Group Chat
- For RCS, the described procedure shall only be initiated when the Contact header field received in the SIP INVITE request or SIP 200 OK response during the setup of the Group Chat included a 3GPP IARI tag carrying *urn:3Aurn-7%3A3gpp-application.ims.iari.rcse.ft*.
- For RCS, a thumbnail may only be included when the Contact header field received during the setup of the Group Chat indicated support for thumbnails as specified in [RCS 5.3] and thumbnail support was enabled on the client as described in [RCS 5.3]
- As disposition notifications only positive-delivery notifications will be requested.

#### 7.4.2 Receiving a CPM File Transfer Request

Following differences with [CPMCONVFUNC]:

- The transfer of multiple files is not applicable for RCS. The reference to the possibility to accept more than one file in step 2, 3, 4, 5 and 6 is thus not relevant in the context of RCS.

### 7.4.3 CPM File Transfer Session Release

Following differences with [CPMCONVFUNC]:

- the steps for multiple files in a session are not applicable for RCS.

### 7.4.4 Fetching a Deferred CPM File Transfer file(s)

No differences with [CPMCONVFUNC].

As clarifications for RCS:

- This procedure shall be initiated on user request as specified in [RCS 5.3].

### 7.4.5 Resuming an interrupted CPM File Transfer

No differences with [CPMCONVFUNC].

#### 7.4.5.1 Requesting Side

Following differences with [CPMCONVFUNC]:

- For step 2, the option to request multiple files is not applicable for RCS.

As clarifications for RCS:

- Failure of the resume procedure is handled as specified in [RCS 5.3].

#### 7.4.5.2 Responding Side

No differences with [CPMCONVFUNC].

As clarifications for RCS:

- If the file is still known and unchanged the resume request may be automatically accepted.

## 8 Procedures at CPM Participating Function

### 8.1 Registration

No differences with [CPMCONVFUNC].

#### 8.1.1 Receive SIP REGISTER Notification

No differences with [CPMCONVFUNC].

#### 8.1.2 Receive Registration Event Information Notifications

No differences with [CPMCONVFUNC].

#### 8.1.3 Terminating the Subscription to Registration Event Information

No differences with [CPMCONVFUNC].

#### 8.1.4 Using the Registration Event Information

No differences with [CPMCONVFUNC].

As a clarification for RCS:



- If the stored messages for chat described in [RCS5-IMENDORS] are handled on the same server this same mechanism may be used to determine that those may be forwarded.

## 8.2 Procedures in the Originating Network

No differences with [CPMCONVFUNC].

### 8.2.1 CPM Standalone Message Handling

#### 8.2.1.1 Handle a Pager Mode CPM Standalone Message

Following differences with [CPMCONVFUNC]:

- Step 2 (Checking of the User-Agent version) is not applicable for RCS
- Step 5 and its sub steps including the NOTE (including external content) are not applicable for RCS
- In step 6 the recording of CPM Conversation History is not dependent on user preferences controlled through XML (extensible Markup Language) Document Management (XDM). For RCS it is always enabled for normal messages.
- In step 8, because step 5 is not applicable for RCS, the size of the message won't be compared to the 1300 bytes limit. The message will always be sent onwards as a pager mode request.
- Step 8 a is not applicable for RCS. The sub steps are applicable though
- Step 9 and its sub steps are not applicable for RCS

As a clarification for RCS:

- This handling includes the processing of delivery and display notifications in which case the handling in section 8.2.4 should also be taken into account
- In step 3 RCS will never allow anonymity and thus reject all messages where it is requested
- For RCS, if a delivery notification and/or read report was requested, the originating participating function will not add its address in an IMDN-Record-Route header
- For RCS, in step 8 a ii the interworking to the ISF will be done on the following error responses unless the message is a delivery or read notification:
  - 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"

NOTE: if a CPM request is about to be routed through a border element to a network with which no CPM interworking agreement exists, that border

element will have to reject the request with one of the above error responses in order to allow for a fallback to other means of delivering the message.

### 8.2.1.2 Handle a Large Message Mode CPM Standalone Message

Following differences with [CPMCONVFUNC]:

- Step 2 of the handling of an INVITE request (Checking of the User-Agent version) is not applicable for RCS
- In step 5 of the handling of an INVITE request, the CPM Participating function will verify that a Session-Expires header is included with the refresher parameter set to “uac”. If this is not the case, the request will be rejected with a 403 “Forbidden” response that will include a warning header with the warning text set to “122 Function not allowed”
- Step 8 of the handling of an INVITE request is not applicable for RCS
- For RCS, the originating participating function will start listening for the incoming MSRP session of the originating client before sending the 200 OK response. That is before step 5 of the handling of a 200 OK response. The MSRP session towards the terminating or controlling function will be set up as soon as the ACK has been sent to the network rather than waiting for the first MSRP SEND from the originating client. That is steps 3 to 5 of the handling of an MSRP SEND request are executed already after forwarding the ACK.
- In step 6 of the handling of a 200 OK response, the recording of CPM Conversation History is not dependent on user preferences controlled through XDM. For RCS it is always enabled.
- For RCS, the same handling as for a SIP BYE will apply when a SIP BYE was received from the controlling or terminating participating function in which case step 3 of the handling of a SIP BYE request will be executed in the dialog towards the originating client.

As a clarification for RCS

- In step 3 RCS will never allow anonymity and thus reject all messages where it is requested
- In case an error response is received from the ISF or an involved IWF (Interworking Function) on a SIP or MSRP request, the response will be forwarded to the client
- For RCS the interworking to the ISF will be done when one of the following error responses is received:
  - 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”
- If a MSRP error response is received, no interworking will be attempted for RCS

NOTE: if a CPM request is about to be routed through a border element to a network with which no CPM interworking agreement exists, that border element will have to reject the request with one of the above error responses in order to allow for a fallback to other means of delivering the message.

## 8.2.2 CPM Session Handling

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- In RCS, the participating function will always stay in the media path.

### 8.2.2.1 Handle a CPM Session Invitation

Following differences with [CPMCONVFUNC]:

- Step 2 of the handling of an INVITE request (Checking of the User-Agent version) is not applicable for RCS
- In step 4 of the handling of an INVITE request also any recipient-list body of the original INVITE request will be included in the generated INVITE request as well as any Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID headers from the original INVITE request. Next to those the INVITE request shall also carry a Supported header with the option tag 'timer' and, if included in the original INVITE request, a 'recipient-list-invite' tag. Furthermore a Session-Expires header with the refresher parameter set to "uac" will be included.
- For RCS, the originating participating function will start listening for the incoming MSRP session of the originating client before sending the 200 OK response. That is before step 6 of the handling of a 200 OK response. The MSRP towards the terminating or controlling function will be set up as soon as the ACK has been sent to the network.
- In step 7 of the handling of a 200 OK response for RCS the session shall always be recorded as defined in [RCS 5.3]. User preferences shall thus not be checked.

As a clarification for RCS

- In step 3 RCS will never allow anonymity and thus reject all messages where it is requested
- In step 4 h of the handling of an INVITE request, for RCS the originating participating function will set the a=setup attribute to the value of "active"
- For RCS, in step 4 of the handling of an INVITE request any Subject header included in the original INVITE request will be included transparently in the generated request.
- Step 5 of the handling of an INVITE request (handling in case the participating function does not stay in the media path) is not applicable for RCS
- In step 1 a of the handling of a 200 OK response, for RCS the service provider policy will never allow to attempt to establish another session for the not accepted media streams
- In step 4 of the handling of a 200 OK response, for RCS the originating participating function will set the a=setup attribute to the value of "passive" according to [RFC6135]

- For RCS the interworking to the ISF will be done when one of the following error responses is received:
  - 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”

NOTE: if a CPM request is about to be routed through a border element to a network with which no CPM interworking agreement exists, that border element will have to reject the request with one of the above error responses in order to allow for a fallback to other means of delivering the message.

### 8.2.2.2 Handle a Cancel Request

No differences with [CPMCONVFUNC].

### 8.2.2.3 Handle a SIP BYE Request

Following differences with [CPMCONVFUNC]:

- For RCS in step 1, if in a CPM Group Session there is no active session with the Controlling Function the CPM Participating Function shall restart the Chat using the procedures defined in section 3.4.4.1.7 of [RCS 5.3] before executing step 1 b to step 1 d.
- For RCS in step 1, the CPM Participating Function shall discard any remaining stored messages and notifications.
- For RCS whether step 1 d ii is applicable depends on local service provider policy rather than on the action element <allow-offline-storage>.
- Step 2 b, 2d and 2 e only apply in case a CPM client is still in the session.
- For RCS if in a CPM Group Session there are still stored messages of the Group Chat to be delivered to the client step 2 b, 2d and 2 e are only executed once those messages are delivered when during this delivery the client performed no action that resulted in the restart of the session.
- For RCS whether step 2 c is applicable depends on local service provider policy rather than on the action element <allow-offline-storage>.
- For RCS whether step 3 c ii is applicable as [RCS 5.3] depends on local service provider policy rather than on the action element <allow-offline-storage>
- For RCS step 4 a-d is not applicable. Whether storage is done shall depend on operator policy. In case storage is performed, step 4 i to iii applies without checking conditions 4 a-d in step 4 iii. Otherwise the SIP BYE request will be handled as described in step 1.
- For RCS step 4 i is only applicable in case the CPM Participating Function does not act as a B2BUA for the subscriptions. Otherwise the leg of the subscription towards

the client is terminated according to the normal procedures in [RFC3265] and [RFC4575] while the leg towards the Controlling Function is maintained.

As a clarification for RCS:

- For RCS, the participating function will always work as a B2BUA
- For RCS, step 2 will apply for all SIP BYE requests that are received on the CPM Participating Function's leg towards the CPM Controlling Function or the other CPM Participating Function involved in the CPM session irrespective of the value of the Reason Header field that is included.
- For RCS, the service provider policy in step 3 b and 3 c will always indicate closure of the session.

#### **8.2.2.4 SIP Session Timer Expiry**

Following differences with [CPMCONVFUNC]:

- For RCS whether step 2 is applicable depends on the local service provider policy rather than on the action element <allow-offline-storage>.

As a clarification for RCS:

- For RCS, the participating function will always work as a B2BUA
- For RCS, only after a response was received on the BYE requests towards a certain party will the participating function release the media plane sessions in step 6

#### **8.2.2.5 Handle a CPM Session Modification Request**

Not Supported for RCS. A session modification request will be rejected with a 488 "Not Acceptable Here" response.

#### **8.2.2.6 Handling of Participants Information**

Following differences with [CPMCONVFUNC]:

- In step 4, substep 4.b is applicable for RCS but not 4.a.
- In step 5, the CPM Participating Function shall include in the Contact header field a URI identifying its own address

As a clarification for RCS:

- For RCS, the CPM participating function will also receive SIP SUBSCRIBE requests initiated by the client since it functions as a B2BUA for the SIP INVITE session. When receiving such a request, the CPM Participating Function shall, when it has no own subscription already (see e.g. what is described in section 8.2.2.6 of [CPMCONVFUNC]), route it onwards to the CPM Controlling Function by resolving the request URI back to the original URI as defined in step 5 of the handling of a SIP 200 OK response in section 8.2.2.1 of [CPMCONVFUNC]. In RCS, the Participating Function receiving a SUBSCRIBE from the RCS client shall either behave as Proxy or as B2BUA. When the CPM Participating Function has a subscription already, the CPM Participating Function shall either connect this subscription back-to-back with the one from the client or terminate its own subscription and route the client's subscription onwards to the Controlling Function.
- For RCS, as an optimisation if there is no established session with the Controlling Function (which should only be the case when forwarding content), the CPM

Participating Function may decide not to forward a SIP SUBSCRIBE request received from the client to the Controlling Function, but instead return a SIP 200 OK response and generate an initial NOTIFY request to the client containing the conference state information based on the stored participant information from the chat.

#### **8.2.2.6.1 Receive Participant Information Notification**

No differences with [CPMCONVFUNC].

#### **8.2.2.7 CPM Group Session Re-join Requests**

##### **8.2.2.7.1 General Re-join Handling**

Following differences with [CPMCONVFUNC]:

- The Contact header is not taken into account as it will have been set to the device's identity.
- For RCS, any session shall be considered to be a CPM Long-Lived Group Session
- If there was already a SIP INVITE request for the same Group Chat sent to the client, the Participating Function shall handle the situation as described in [RCS 5.3].
- For RCS, the INVITE request is only forwarded to the controlling function when the receiving client actually sends a message.

As a clarification for RCS:

- As the CPM Group Session Identity has been determined by the CPM Participating Function itself, the CPM Participating Function is expected to be able to differentiate it from a normal user or other identity.

##### **8.2.2.7.2 CPM Long-lived Group Session**

Following differences with [CPMCONVFUNC]:

- In step 1 when the session is inactive, the CPM client shall always be considered to be allowed to restart the chat session regardless of the domain of the Controlling Function hosting the focus.
- Step 2 when the session is inactive is not applicable for RCS
- Step 2 when the session is active is not applicable for RCS.

#### **8.2.2.8 Handling of Failed Deferred Delivery of Chat Messages**

No differences with [CPMCONVFUNC].

#### **8.2.3 CPM File Transfer Handling**

##### **8.2.3.1 Handle a CPM File Transfer**

Following differences with [CPMCONVFUNC]:

- For RCS, the CPM Participating Function shall not check the OMA-XDM-Policy, but rather apply a Service Provider Policy on the maximum allowed File Transfer size. In case the file-selector:size attribute exceeds the value determined by this Service Provider Policy the handling shall be as described in [CPMCONVFUNC] for the situation where it exceeds the value set in the OMA-XDM-Policy.

- In step a, the URI in the Contact Header field shall be set to a URI identifying the own address of the CPM Participating Function

As a clarification for RCS:

- For RCS the file transfer request will never be routed to the ISF
- The participating function will always remain in the media path for a file transfer
- All relevant differences and clarifications described in chapter 8.2.2.1 of this document are also applicable to file transfer

### **8.2.3.2 Handle a CPM File Transfer Closing Request**

Following differences with [CPMCONVFUNC]:

- For RCS step 1 will not depend on the action element <allow-offline-storage>, but rather on service provider policies

As a clarification for RCS:

- For an RCS file transfer request no IWFs will have been involved
- The participating function will always remain in the media path for a file transfer
- All relevant differences and clarifications described in chapter 8.2.2.3 of this document are also applicable to file transfer

### **8.2.3.3 Handling of Failed Deferred Delivery of File Transfers**

No differences with [CPMCONVFUNC].

## **8.2.4 Receiving a Disposition Notification**

No differences with [CPMCONVFUNC].

## **8.2.5 Sending a Disposition Notification**

Not applicable for RCS as the originating Participating Function does not have to generate disposition notifications itself.

## **8.2.6 Media Handling by the Originating Participating Function**

Following differences with [CPMCONVFUNC]:

- Step 3 (URL indirection) is not applicable for RCS;
- For RCS, after step 6 if a delivery notification and/or read report was requested, the originating participating function will not add its address in an IMDN-Record-Route header.

## **8.3 Procedures in the Terminating Network**

No differences with [CPMCONVFUNC].

### **8.3.1 CPM Message Handling**

#### **8.3.1.1 Handle a Pager Mode CPM Standalone Message and SIP IMDNs**

Following differences with [CPMCONVFUNC]:

- Step 1 (checking of the User-Agent version) is not applicable for RCS

- Step 3 (comparison to the standalone messages Blacklist URI-list) is not applicable for RCS if personal network blacklist functionality is not enabled
- The handling in step 4 will not depend on the recipient's user preferences, but will rather follow a fixed policy which will be the same for all requests. This policy will either have
  - A rule including the equivalent of <allow-delivery-and-interwork> and of a <allow-offline-storage> sub-element of the action element set to "true" for the CPM enabled RCS users that also have non-CPM clients (e.g. an RCS-e 1.2 client). This rule shall not be used for delivery reports and display notifications however.
  - A rule including the equivalent of <allow-offline-storage> sub-element of the action element set to "true" for the CPM enabled RCS users that have no non-CPM clients  
This rule shall always apply for messages carrying delivery reports and display notifications
  - A rule including the equivalent of <allow-interwork> sub-element of the action element for all non-CPM users (regardless of whether they are RCS enabled or not).
- Steps 4 a, 4 c, 4 d and 4 e are thus not applicable for RCS
- Step 5 (support for a "do not disturb" rule) is not applicable for RCS
- In step 6 used rules are not dependent on user preferences controlled through XDM. For RCS it's always enabled for CPM users. This would be the equivalent of having a rule matching all requests with the "<allow-offline-storage>" sub-element of the "action" element set to "true". See the described difference for step 4. This behaviour does not apply for SIP MESSAGE requests carrying a delivery or read notification
- Step 8 a i is not applicable for RCS. This means that no registered CPM client will be excluded.
- Step 8 a iii is not applicable for RCS. That is no other information sources will be consulted to decide on the delivery of the request apart from the registration information
- Step 8 a iv is not applicable for RCS. All RCS clients should have the capability to receive all CPM Standalone Messages
- Step 8 b (replacing media with a reference) is not applicable for RCS
- In step 8 f, the GRUU will be included only if the network does support GRUU and the public GRUU is not included in the request already. If GRUU is not supported by the network a sip.instance parameter will be included as described in section 6.1.2.
- Step 2 when no 200 OK was received is not applicable for RCS. The service provider policy will always indicate to defer the message

#### As a clarification for RCS

- In step 2 RCS will never allow anonymity and thus reject all messages where it is requested
- For RCS, if personal network blacklist functionality is enabled the list used in step 3 will be *rcs\_pnb\_standalone\_blockedusers* as described in [RCS 5.3].
- For RCS, if a delivery notification and/or read report was requested, the terminating participating function will not add its address in an IMDN-Record-Route header



- For RCS, in step 8 an equivalent rule with only the <allow-offline-storage> sub-element of the action element is not considered to have matched in step 4
- In Step 8 a v according to service provider policy, a message may not be delivered to a client if the client is known to be roaming (based on the SIP REGISTER information obtained according to chapter 8.1) and/or the message is larger than an operator configurable limit. Furthermore if the received message included a GRUU in the request URI (that is it is a delivery or display notification), all clients for which those identifiers do not match will be excluded.
- As this is fixed policy which is not controlled by the clients, it is out of scope for RCS how an implementation should achieve this. It could implement this fixed policy internally in the participating function, but it might also choose to rely on the XML Document Management Server (XDMS) which would require the use of pre-provisioned documents, as they are not managed by the RCS clients.

### 8.3.1.2 Handle a Large Message Mode CPM Standalone Message

Following differences with [CPMCONVFUNC]:

- Step 1 (Checking of the User-Agent version) is not applicable for RCS
- Step 5 (comparison to the standalone messages Blacklist URI-list) is not applicable for RCS if personal network blacklist functionality is not enabled
- The handling in step 6 will not depend on the recipient's user preferences, but will rather follow a fixed policy which will be the same for all requests. This policy will either have
  - An equivalent rule including the <allow-delivery-and-interwork> and a <allow-offline-storage> sub-element of the action element set to "true" for the CPM enabled RCS users that also have non-CPM clients (e.g. an RCS-e 1.2 client).
  - An equivalent rule including the <allow-offline-storage> sub-element of the action element set to "true" for the CPM enabled RCS users that have no non-CPM clients
  - An equivalent rule including the <allow-interwork> sub-element of the action element for all non-CPM users (regardless of whether they are RCS enabled or not).

Note: depending on implementation this last rule may be used in combination with the equivalent of <allow-offline-storage> sub-element
- Steps 6 a, 6 c, 6 d and 6 e are thus not applicable for RCS
- In step 6 b i and 8 b of the handling of an INVITE request and step 2 when no 200 OK response was received also any recipient-list-history body of the original INVITE request will be included in the generated INVITE request as well as any Subject, Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID headers from the original INVITE request. Next to those the INVITE request will also include a Supported header with the option tag 'timer' and a 'recipient-list-invite' tag if included in the original INVITE request. Furthermore a Session-Expires header with the refresher parameter set to "uac" in case of step 6 b i and to "uas" in case of step 8 b will be included. In step 2 when no 200 OK response was received the refresher parameter in the Session-Expires header will be set to "uac"
- Step 7 (support for a "do not disturb" rule) is not applicable for RCS

- In step 8 a, the differences and clarifications for RCS that are described to chapter 8.3.1.1 in this document are applicable as well.
- In step 8 e, the GRUU will be included only if the network does support GRUU and the public GRUU is not included in the request already. If GRUU is not supported by the network a sip.instance parameter will be included as described in section 6.1.2.
- The execution of the processing described in chapter 8.5 is not dependent on user preferences controlled through XDM. For RCS it's always enabled for CPM users. This would be the equivalent of having a rule matching all requests with the "<allow-offline-storage>" sub-element of the "action" element set to "true". See the described difference for step 6
- In the handling of a 200 OK response for RCS, the terminating participating function will start listening for the incoming MSRP session of a CPM client already before sending it the ACK request. That is before step 1 of the handling of a 200 OK response. An MSRP session towards the ISF/IWF will be set up after the ACK has been sent towards it as described in step 2.
- In the handling of a 200 OK response for RCS, in step 4 a BYE will only be generated after one was received from the originating side and all chunks received far on the user plane has been transmitted towards the sender of the 200 OK response.
- In case a BYE is received from a terminating client, the media plane resources to that recipient will be released and a 200 OK response will be sent to that client.
- Step 1 when no 200 OK was received will also apply in case BYE requests were received from all clients that originally accepted the message
- Step 2 and 3 when no 200 OK was received is not applicable for RCS. The service provider policy will always indicate to defer the message

#### As a clarification for RCS

- In step 2 RCS will never allow anonymity and thus reject all messages where it is requested
- In step 3, for RCS the participating function shall handle the header according to [RFC4028]
- For RCS, if personal network blacklist functionality is enabled the list used in step 3 will be *rcs\_pnb\_standalone\_blockedusers* as described in [RCS 5.3].
- In Step 3 of the handling of a 200 OK response, for RCS the timing of the forwarding will depend on Local Server Policy<sup>1</sup>. For RCS this policy will as well as allowing the complete re-assembly of the received message before starting the forwarding also allow to forward in step 3 all chunks that have been received so far in case all CPIM headers have been received already. Otherwise with this setting of the policy forwarding should begin as soon as all CPIM headers are received. Again with this policy all further chunks will be forwarded as soon as the MSRP headers have been completely received.
- For RCS, if a delivery notification and/or read report was requested, the terminating participating function will not add its address in an IMDN-Record-Route header

---

<sup>1</sup> Several policies can be available depending on operator requirements: the specified behaviour is highly recommended when there are no screening requirements: it minimizes buffering delays on the Participating Function.

### 8.3.1.3 Applying delivery policies

No differences with [CPMCONVFUNC].

### 8.3.1.4 Replacing Media with a Reference

Not applicable for RCS

### 8.3.1.5 Establish MSRP Session for Receiving Large Message Mode CPM Standalone Message

Following differences with [CPMCONVFUNC]:

- The 200 “OK” response to be generated is in response to the INVITE request received in chapter 8.3.1.2 and will contain the Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID headers from the original INVITE request. Next to those also a Supported header with the option tag ‘timer’ and a Session-Expires header with the refresher parameter set to “uas” will be included.
- For RCS, the terminating participating function will start listening for the incoming MSRP session from the originating side already before sending a 200 OK response. That is before step 5 of the establishment of an MSRP session
- A received CANCEL request will also be forwarded to all clients and/or the ISF in case there is still an outstanding INVITE request to that entity
- For RCS step 1 of receiving a MSRP SEND will be subject to Local Server Policy<sup>2</sup>. For RCS this policy will also offer to forward immediately any received chunks to any terminating entity. That is either the ISF or a CPM client to which a session was setup for the transmission of this standalone message. This policy would allow this as soon as all CPIM headers have been received and from then on as soon as the MSRP headers are received for a certain chunk.

As a clarification for RCS:

- For RCS, if a delivery notification and/or read report was requested, the terminating participating function will not add its address in an IMDN-Record-Route header

### 8.3.1.6 Defer CPM Standalone Messages

Following differences with [CPMCONVFUNC]:

- For RCS, both the behaviour in which there is a n equivalent rule in which the “<action>” element includes a “<allow-push>” sub element set to “true” and the behaviour in which there is a rule in which the “<action>” element includes a “<allow-pull>” sub element set to “true” is applicable.
- Case 3 is not applicable

As a clarification for RCS:

- As this is fixed behaviour which is not controlled by the clients, it is out of scope for RCS how an implementation should achieve this. It could implement this fixed policy internally in the participating function, but it might also choose to rely on the XDMS

---

<sup>2</sup> Several policies can be available depending on operator requirements: the specified behaviour is highly recommended when there are no screening requirements: it minimizes buffering delays on the Participating Function.

which would require the use of pre-provisioned documents including this rule, as they are not managed by the RCS clients.

#### **8.3.1.6.1 Storing a Deferred CPM Message in the Deferred CPM Message Queue**

Following differences with [CPMCONVFUNC]:

- For a Large Message Mode request also the message size and the accept-wrapped-types as provided in the SDP will be stored.

#### **8.3.1.6.2 Pushing Deferred CPM Messages**

Following differences with [CPMCONVFUNC]:

- The participating function will not check whether there is a relevant “do-not-disturb” rule in the [OMA-XDM-Policy]
- When a client comes online, the do-not-disturb flag in the user preferences will not be checked
- For RCS, if a client that comes online is considered not to be suitable the message will remain deferred, allowing the client to retrieve it through pull
- For RCS, if the participating function has information that a user didn’t come online, but can receive a message provided that it is interworked, the participating function can decide to push the deferred messages to the ISF. How the participating function obtains this information is out of the scope of this document. The participating function can do this as soon as the message is deferred or later on when it finds out that the user’s capability for receiving a message that is interworked has changed.

As a clarification for RCS:

- The CPM Participating Function will not send out-of-band notifications

#### **8.3.1.6.3 Sending Notifications and Awaiting CPM Client Action**

Following differences with [CPMCONVFUNC]:

- The participating function will not check whether there is a relevant “do-not-disturb” rule in the [OMA-XDM-Policy]
- In step 1 for sending a notification, step 8 a v of section 8.3.1.1 will not be considered when evaluating whether a client is suitable as otherwise a message deferred due to such an operator policy cannot be retrieved. That means that all clients that are registered are considered suitable.
- Step 3 for sending a notification is not applicable for RCS. No out-of-band notifications will be sent
- In the NOTE1 the case for out-of-band notifications is not applicable for RCS
- Step 4 when handling a received INVITE request is not applicable for RCS
- Step 5 when handling a received INVITE request is not applicable for RCS

As a clarification for RCS

- In step 1 e: the CPMDeferredMsgMgmtURI will be provisioned to the clients as described in [RCS 5.3].
- When handling a SIP INVITE request, the Request-URI will be set to the value provisioned to the clients for CPMDeferredMsgMgmtURI as described in Annex A of [RCS 5.3].
- For RCS only summary information will be sent

#### **8.3.1.6.4 Delivering Deferred CPM Messages to the Message Storage Server**

Not applicable for RCS

#### **8.3.1.6.5 Handle a Deferred CPM Message Information Request**

Following differences with [CPMCONVFUNC]:

- In Step 2, the response will contain an Expires header with the value of 0
- In Step 5, the NOTIFY request will include a Subscription-State header set to "terminated "

#### **8.3.1.6.6 Sending a Pager Mode Deferred CPM Message**

Following differences with [CPMCONVFUNC]:

- In Step 5, if the network supports GRUU the Request URI shall include the GRUU of the targeted client.
- In Step 7, the ICSI "3gpp-service.ims.icsi.oma.cpm.msg " out of the stored Accept-Contact will not be included as the "3gpp-service.ims.icsi.oma.cpm.deferred" is already there
- In step 8 the recording shall always be enabled for RCS.
- In step 11, for RCS the SIP MESSAGE request may instead be sent to the ISF if the participating function decided to push the message to the ISF after it obtained information that the recipient of the message can receive it if it is interworked. The means for the participating function to obtain this information are outside of the scope of this document.

As a clarification for RCS

- In step 7, if a delivery notification and/or read report was requested, the participating function will always add it's address in an IMDN-Record-Route header
- In step 9, the media object will never be replaced with a reference for RCS

#### **8.3.1.6.7 Sending a Large Message Mode Deferred CPM Message**

Following differences with [CPMCONVFUNC]:

- Also the Conversation-ID, Contribution-ID and if stored the recipient-list-history body, Subject and InReplyTo-Contribution-ID headers out of the original INVITE request will be added to the generated INVITE request.
- In Step 5, the Request URI will be set to the GRUU of the targeted client if the network supports GRUU. Otherwise an Accept-Contact header with the sip.instance tag and device identifier of the targeted client will be included as defined in section 6.1.2 of [CPMCONVFUNC].
- In step 8, the refresher parameter will be set to "uas"
- In step 9, also a a=accept-wrapped-types attribute will be included with the same value as the one out of the original request.
- In step 9 d, the size will be set to the size that was stored from the original request
- In step 10, for RCS the SIP INVITE request may instead be sent to the ISF if the participating function decided to push the message to the ISF after it obtained information that the recipient of the message can receive it if it is interworked. The means for the participating function to obtain this information are outside of the scope of this document.

- Step 3 of the handling of a 200 OK response will be done before step 2. That is the participating function will start listening for an incoming MSRP session before sending the ACK.
- Before executing Step 5 of the receiving of a 200 OK response, the participating function will wait until the client has set up a MSRP session for receiving the message.
- In step 1 after the message was successfully transferred, for RCS the preference will always be to store the CPM Conversation History
- After a response was received to the BYE request send in step 2 after the message was successfully transferred, the media plane will be released.
- If an error response is received on the INVITE request or on one of the MSRP SEND requests, the message will not be stored and remain in the deferred CPM message queue.
- If an error response is received on the INVITE request, an ACK request will be send to acknowledge its reception
- If an error response is received on one of the MSRP SEND requests or the TCP connection used for the MSRP session is lost, a BYE request will be sent and when a response has been received to that the resources on the media plane will be released

As a clarification for RCS

- In step 4 of the handling of a 200 OK response, the media object will never be replaced with a reference for RCS
- In step 5, if a delivery notification and/or read report was requested, the participating function will always add its address in an IMDN-Record-Route header

#### **8.3.1.6.8 Handling Deferred CPM Messages on Expiry Time**

Following differences with [CPMCONVFUNC]:

- For RCS, the preference retrieved in step 1 will always be set to “discard”. As this is fixed behaviour which is not controlled by the clients, it is out of scope for RCS how an implementation should achieve this. It could implement this fixed policy internally in the participating function, but it might also choose to rely on the XDMS which would require the use of pre-provisioned documents, as they are not managed by the RCS clients.
- Step 2 a ii (sending of notifications that the deferred message was discarded) is not applicable for RCS
- Step 2 b (a user preference set to “store”) is not applicable for RCS

As a clarification for RCS:

- Step 2 a iii (sending of negative delivery reports) is not applicable for RCS as RCS clients will not request such reports

#### **8.3.2 CPM Session Handling**

No differences with [CPMCONVFUNC].

As a clarification for RCS: the participating function will always stay in the media path.

### 8.3.2.1 Handle a CPM Session Invitation

Following differences with [CPMCONVFUNC]:

- Step 1 of the handling of an INVITE request (Checking of the User-Agent version) is not applicable for RCS
- Step 4 of the handling of an INVITE request (checking of do-not-disturb rule) is not applicable for RCS
- For RCS, a CPM Participating Function providing deferred delivery functionality for Group Chat may after step 5 return a 200 OK response and establish the MSRP session. When the user would reject in the invitation with a 603 Decline response, a CPM Participating Function that accepted the session in this way shall discard any stored messages and terminate the session by sending a SIP BYE request carrying a Reason header field with the protocol set to SIP and the protocol\_cause to 200 (e.g SIP;cause=200;text="Call completed") to the CPM Controlling Function.
- In step 8 d vii of the handling of an INVITE request also any recipient-list body of the original INVITE request will be included in the recipients-list-history of the generated INVITE request as per [RFC5364].
- Step 9 of the handling of an INVITE request (handling of the INVITE request when functioning as a proxy) is not applicable for RCS
- In step 4 of the handling of the first 200 OK response when acting as a B2BUA the a=setup attribute in the SDP will be set to the value of "passive"
- The CPM Participating function will begin to listen for an incoming MSRP session from the originating participating or controlling function before step 6 of the handling of the first 200 OK response when acting as a B2BUA
- For RCS step 8 of the handling of the first 200 OK response when acting as a B2BUA is applicable only subject to local policy
- The handling of the first 200 OK response when acting as a SIP Proxy is not applicable for RCS

As a clarification for RCS

- In step 2 of the handling of an INVITE request RCS will never allow anonymity and thus reject all messages where it is requested
- For RCS, if personal network blacklist functionality is enabled the list used in step 3 will be *rcs\_pnb\_chat\_blockedusers* as described in [RCS 5.3].
- In step 6 and step 8 b of the handling of an INVITE request all registered RCS clients are considered suitable
- In Step 6 of the handling of an INVITE request the service provider policy will be to deliver the invitation through a non-CPM service in case the addressed user is not a CPM user. That is step 6 b. In case the user is a CPM user, the session invitation will be rejected. That is step 6 a.

#### 8.3.2.1.1 Handling of CPM Session responses from CPM Clients

Following differences for RCS:

- For step 1 when receiving a SIP Final Response is applicable for RCS when all the RCS clients responded with other than 200 OK, rather than 'any' of the clients.
- Step 1 when receiving a SIP Final Response other than 200 OK from all clients (i.e. a SIP 302 response), is not applicable for RCS

- When executing step 2 when receiving a SIP final response other than 200 OK from all clients and if the session is a Group Session, the Participating Function shall also subscribe to the CPM Group Session Participant Information on behalf of the served CPM User.
- In step 3 when receiving a SIP Final Response other than 200 OK from all clients, the service provider policy for RCS will indicate that no interworking is done. Step 3 ii is therefore not applicable for RCS
- Before forwarding a received ACK request to a 200 OK response to the client that accepted the session, the participating function will start listening for an incoming MSRP session from that client
- For RCS, if the ISF did not accept all media in the session, no attempt will be made to establish a new session for the remaining media. Steps 1 and 2 in the handling of a 200 OK response from the ISF are thus not applicable for RCS.
- For RCS when in a Group Session there is an INVITE request for the same Group Chat sent to the Conference Focus, the case shall be handled as described in [RCS 5.3].

#### **8.3.2.1.2 MSRP Media Stream with one CPM Client**

Not applicable to RCS

#### **8.3.2.1.3 Multiple MSRP Media Streams**

No differences with [CPMCONVFUNC].

#### **8.3.2.1.4 Media Plane Handling**

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- The procedures apply the same to each client of the RCS user;
- In step 2 when receiving an MSRP SEND is applicable in RCS according to service provider policy rather than based on user preferences.
- Step 4 should be applicable even if a session is set up with only one of the clients.

#### **8.3.2.2 Handle a Cancel Request**

No differences with [CPMCONVFUNC].

#### **8.3.2.3 Handle a SIP BYE Request**

Following differences with [CPMCONVFUNC]:

- same differences as described in section 8.2.2.3 "*Handle a SIP BYE Request*".

As a clarification for RCS:

- For RCS, the participating function will always work as a B2BUA

#### **8.3.2.4 SIP Session Timer Expiry**

Following differences with [CPMCONVFUNC]:

- For RCS step 1 is only applicable if recording of sessions is enabled based on local policy



As a clarification for RCS:

- For RCS, the participating function will always work as a B2BUA

### **8.3.2.5 Handle a CPM Session Modification Request**

Not supported for RCS. A session modification request will be rejected with a 488 “Not Acceptable Here” response.

### **8.3.2.6 Handling of Participants Information**

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- The differences and clarifications provided in section 8.2.2.6 apply for the terminating Participating Function as well.

### **8.3.2.7 Receive Participant Information Notification**

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- The differences and clarifications provided in section 8.2.2.6.1 apply for the terminating Participating Function as well.

### **8.3.2.8 CPM Group Session Re-join Requests**

Not applicable for RCS as re-join is an originating side function only

### **8.3.2.9 Deferred CPM Session handling**

No differences with [CPMCONVFUNC]:

As a clarification for RCS:

- For RCS, the procedures apply to 1-to-1 Chat as well with the exception of the subscription to obtain the latest participant information and the handling of the rejoin.
- RCS clients shall support always IMDN.

#### **8.3.2.9.1 Deferred delivery initiated by the CPM Participating Function**

##### **8.3.2.9.1.1 1-1 CPM Session delivery**

Following differences with [CPMCONVFUNC]:

- For RCS in step 6, the device identifier of the device that originally sent the message(s) shall be included.
- For RCS step 3 when performing the delivery of IMDNs is not applicable as notifications from multiple devices could be delivered in a single session
- For RCS step 5 when performing the delivery of IMDNs is not applicable. Instead the P-Asserted-Identity header field shall be set to a URI with the username part set to “rcse-standfw” and the domain to the operator’s domain.

As a clarification for RCS:

- As specified in [RCS 5.3], in step 1 depending on service provider policy a first message may be included as a CPIM body part in this SIP INVITE request.

- For RCS in step 11 a. ii. and 4 of receiving 200 “OK” to the SIP INVITE request, as specified in [RCS 5.3] a following MSRP SEND request shall only be sent when a MSRP 200 OK response has been received to the previous one.
- For RCS, all CPM clients shall support generation of IMDNs in 1-to-1 Chat. The disposition notifications are therefore never generated by the CPM Participating Function.

#### **8.3.2.9.1.2 CPM Group Session delivery**

Following differences with [CPMCONVFUNC]:

- For RCS, the session will always be considered to be active. The RCS Participating Function shall ensure that an active session is available by performing the procedures described in section 3.4.4.1.7 of [RCS 5.3] whenever the client sends content or a request (adding of participant, explicit departure) that requires an active session.
- For RCS in step c 1 if no active session exists with the CPM Controlling Function the CPM Participating Function shall restart the session by performing the procedures described in section 3.4.4.1.7 of [RCS 5.3] before executing the procedures in step c 1 i and ii.
- For RCS the INVITE request in step a) shall be composed based on the stored data and thus for example include the isfocus parameter in the Contact header field.
- For RCS in step d iii 1. the CPM Participating Function shall either unsubscribe from the conference state as soon as a SUBSCRIBE request for conference state is received from the client or connect the subscription from the client as a B2BUA with the own subscription.
- For RCS step d iii 1. the CPM Participating Function shall only connect the legs as described in section 8.3.2.8

As a clarification for RCS:

- For RCS, when the CPM Participating Function has knowledge that the CPM client lost connectivity while the session was active (e.g. because it received a SIP BYE request from the client side carrying a Reason header field with the protocol set to SIP and with the protocol\_cause being a value different from 200), the CPM Participating Function may delay the deferred delivery of CPM Group Session data until after an implementation dependent timeout to avoid the race condition between this deferred delivery and the automatic rejoin performed by the client.

#### **8.3.2.9.1.3 Deferred delivery performed upon CPM Client re-join**

No differences with [CPMCONVFUNC].

#### **8.3.2.9.1.4 Deferred delivery race conditions**

Not applicable for RCS. In RCS the CPM Participating Function will accept the rejoin and rely on the client handling for managing multiple sessions described in [RCS 5.3].

### **8.3.3 CPM File Transfer Handling**

#### **8.3.3.1 Handle a CPM File Transfer**

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- For RCS the file transfer request will never be routed to the ISF
- The participating function will always remain in the media path for a file transfer
- All relevant differences and clarifications described in chapter 8.3.2.1 of this document are also applicable to file transfer
- If the setup of the session fails as described in section 8.3.3.3 of [CPMCONVFUNC] and based on local policy deferred File Transfer handling should be provided, the procedure in section 8.3.3.3 shall be applicable.

#### **8.3.3.2 Handle a CPM File Transfer Closing Request**

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- For a RCS file transfer request no IWFs will have been involved

#### **8.3.3.3 Handling Deferred CPM File Transfer File(s)**

Following differences with [CPMCONVFUNC]:

- For RCS in step 1 c, 2 b and 3 b only a single file shall be accepted.

As a clarification for RCS:

- For RCS when deferred File Transfer is to be provided the CPM Participating Function shall always stay in the media path

#### **8.3.3.4 Delivering deferred CPM File Transfer file(s):**

Following differences with [CPMCONVFUNC]:

- For RCS the procedure in case 3 is not applicable as in RCS this functionality is only provided at the originating side.

#### **8.3.3.5 Generating SIP INVITE request for deferred delivery of CPM File Transfer file(s)**

Following differences with [CPMCONVFUNC]:

- For RCS in step 9 the SDP shall include the file-selector and filetransfer-id attributes that were stored from the original SIP INVITE request as described in section 8.3.3.7.

#### **8.3.3.6 Receiving SIP INVITE request for pull of CPM File Transfer file(s)**

No differences with [CPMCONVFUNC].

#### **8.3.3.7 Storing CPM File Transfer Deferred Information**

No differences with [CPMCONVFUNC].

#### **8.3.3.8 Timing between originating and terminating Participating Function**

No differences with [CPMCONVFUNC].

### 8.3.3.9 Media Plane Handling for MSRP Session

For RCS this procedure is not applicable. Instead the Participating Function shall establish the session acting as passive endpoint, receive the file over this MSRP connection and store it along with the metadata described in section 8.3.3.7. This implies that when connecting to an IWF, the Participating Function will be the passive endpoint and the IWF the active one.

### 8.3.4 Sending a Disposition Notification

Following differences with [CPMCONVFUNC]:

- A disposition notification will also be generated in case one is received from one of the entities to which the message was delivered. That is either the ISF or a CPM client. In case the message was delivered to multiple endpoints a similar caching strategy will be followed as at the originating side. See chapter 8.2.4.
- The storage is based on service provider policy rather than user preferences.
- The sending of a delivery notification in case of an error in the MSRP session of a CPM File Transfer request is for RCS applicable only if a negative-delivery-notification was requested and no deferred handling for File Transfer is provided.

As a clarification for RCS:

- Case 1 and 3 on when to send a disposition notification are not applicable for RCS.

### 8.3.5 Receiving a Disposition Notification

Following differences with [CPMCONVFUNC]:

- Display notifications are not used in RCS for file transfer.

## 8.4 User Preferences

No differences with [CPMCONVFUNC].

As a clarification for RCS

- This section is only applicable in case an implementation chooses to rely on pre-provisioned XDM documents rather than implementing the fixed policies for RCS in the participating function itself.

### 8.4.1 Retrieving User Preferences

No differences with [CPMCONVFUNC].

As a clarification for RCS

- This section is only applicable in case an implementation chooses to rely on pre-provisioned XDM documents rather than implementing the fixed policies for RCS in the participating function itself.

## 8.5 Record CPM Conversation History

No differences with [CPMCONVFUNC].

### 8.5.1 Record CPM Standalone Message

No differences with [CPMCONVFUNC]. Step 4 of [CPMCONVFUNC] is an indentation/typo and shall be applied as a standalone paragraph, applicable to both pager mode and large message mode.

### 8.5.2 Record CPM Session

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- For RCS storing parameters from the SDP is mandatory.
- For RCS in a multi-device setup content shall be recorded live instead of at the end of the session.

### 8.5.3 Record CPM File Transfer

No differences with [CPMCONVFUNC].

As a clarification for RCS:

- Parameters from the SDP shall be stored.

## 9 Procedures at CPM Controlling Function

Following differences with [CPMCONVFUNC]:

- In the third bullet (handling of a CPM session invitation), the procedure will rather be used for the CPM Controlling Function than for the IWF
- The fourth bullet (handling of a CPM File Transfer), the procedure will rather be used for the CPM Controlling Function than for the IWF.

### 9.1 CPM Standalone Message Handling

#### 9.1.1 Pager Mode CPM Standalone Message Handling

Following differences with [CPMCONVFUNC]:

- Step 2 is not applicable to RCS
- Step 4 is not applicable for RCS. Any request sent to a pre-defined group reaching the controlling function will be rejected with a 404 Not Found response. That is the execution of step 4 b without doing the related checks in step 4.
- Step 5 e is not applicable for RCS
- In step 5 g the resource-list body will not be copied from the received SIP MESSAGE request into the outgoing SIP MESSAGE request. Instead a new resource-list body will be included containing the addresses that had "To" and "CC" qualifiers in the resource-list body of the received SIP MESSAGE. In addition, those qualifiers will be copied in the resource-list body and the Require header of the outgoing MESSAGE request will not mention "recipient-list-message". Instead "recipient-list-message" will be included in the Supported Header.
- In step 6 also the Conversation-ID, Contribution-ID will be copied from the received SIP MESSAGE request into the outgoing SIP 202 Accepted Response and a Server header will be included.

As a clarification for RCS:

- In step 3 a, for RCS anonymity will never be allowed
- For RCS, if a delivery notification and/or read report was requested, the CPM Controlling Function will always add its address in an IMDN-Record-Route header in the CPIM headers of the outgoing SIP MESSAGE requests generated in step 5

- For RCS the interworking to the ISF will be done for a group member when one of the following error responses is received on the MESSAGE request sent in step 5 to that party:
  - 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”

### 9.1.2 Large Message Mode CPM Standalone Message Handling

Following differences with [CPMCONVFUNC]:

- Step 2 is not applicable to RCS
- Step 4 is not applicable for RCS. Any request reaching the controlling function will be rejected with a 404 Not Found response. That is the execution of step 4 a without doing the related checks.
- In step 5 of the handling of an INVITE request, the CPM Controlling Function will verify that a Session-Expires header is included with the refresher parameter set to “uac”. If this is not the case, the request will be rejected with a 403 “Forbidden” response that will include a Warning header with the warning text set to “122 Function not allowed”
- In step 7 b also the conversation-ID, contribution-ID and if applicable the Reply-To, Expires, InReplyTo-Contribution-ID and Subject headers will be copied from the received SIP INVITE request into the outgoing SIP INVITE request. Also a resource-list history body will be added to the outgoing SIP INVITE request with the addresses for which the “TO” and “CC” qualifiers were set in the received INVITE request for which also those qualifiers will be copied
- Step 7 h is not applicable for RCS
- In Step 7 j ii the case where privacy was requested is not applicable for RCS
- Step 7 k is not applicable for RCS
- The same handling for when a first 200 OK response is received will be applied in case of a prior 2xx response. Any 200 OK response following after that, would not be treated as the first 200 OK response.
- In step 1 when the first 200 OK response has been received also the Conversation-ID, Contribution-ID will be copied from the received SIP INVITE request into the outgoing SIP INVITE response and a User-Agent header will be included. Also Require and Supported headers with the value “timer” will be included as well as a Session-Expires header with the refresher parameter set to “uac”
- Prior to executing step 5 when the first 200 OK response has been received the CPM controlling function will start listening for incoming MSRP sessions from the party to which the response will be sent

- In Step 2 when receiving a CANCEL request, the reference to a pre-defined group members is not applicable to RCS, but the same handling will be applied to all members of an ad-hoc group
- On any non-2xx type of Response, the controlling function will send an ACK request immediately.
- In case for a 2xx type of response is received from one of the invited members and an ACK request from the inviting party was received and processed already, an ACK request will be generated and sent to the party that sent the 2xx response. After that Step 2 of the handling of a received ACK request will be executed towards the party that sent the 2XX type of response and any MSRP requests stored in step 2 of the handling of an MSRP SEND request will be delivered to that party.
- Step 2 of the handling of a SIP BYE request will only be executed once all remaining data has been sent to that party
- Step 3 of the handling of a SIP BYE request will only be executed after a final response was received to the BYE sent in step 2

As a clarification for RCS:

- In step 3 a, for RCS anonymity will never be allowed
- For the NOTE in step 2 handling of an MSRP SEND request, for RCS the MSRP contents received will be stored until a final SIP response to the SIP INVITE request has been received from all invited group members. A duplicate MSRP SEND request will be delivered to all group members that have responded to the INVITE request with a 2xx type of response.
- In the processing of an MSRP SEND request, the forwarding of the request won't start until the MSRP headers have been received completely and for the initial request(s) also the CPIM headers
- For RCS, if a delivery notification and/or read report was requested, the controlling function will always add its address in an IMDN-Record-Route header in the CPIM headers of the outgoing MSRP SEND requests generated in step 2 of the handling of an MSRP SEND request
- For RCS the interworking to the ISF will be done for a group member when one of the following error responses is received on the INVITE request sent in step 7 to that party:
  - 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"

## 9.2 CPM Group Session Handling

The CPM Group Session Identity is the focus Session Identity in RCS.

### 9.2.1 CPM Group Session Initiation

Following differences with [CPMCONVFUNC]:

- In step 3 if the unsupported SDP parameters are related to the content types in accept-wrapped-types, a SIP 488 “Not Acceptable here” will only be sent in case all included content types are not acceptable. Otherwise only the unsupported content types will be removed from the SDP that is used in the further processing. The CPM token defined for the a=chatroom attribute is used with no differences.
- Step 5 is not applicable for RCS. Any request sent to a pre-defined group reaching the controlling function will be rejected with a 404 “Not Found” response. That is the execution of step 5 a without doing the related checks.
- When receiving a SIP final response other than 2xx from an invited CPM group member, the CPM Controlling Function will not remove the inviting CPM group member from the CPM group session but rather for the ‘invited’ CPM participant (typo). No action will be taken apart from sending an ACK request to the invited CPM group member that sent the response.
- In step 1 when the first 200 OK response has been received also the conversation-ID, contribution-ID will be copied from the received SIP INVITE request into the outgoing SIP INVITE response. Also Require and Supported headers with the value “timer” will be included
- Rather than including a SDP body based on the received SDP answer in step 2 of handling the first 200 “OK” response, the included SDP would be based on the SDP sent by the inviting CPM group member possibly updated with the restrictions imposed in step 3 of the handling of the INVITE request.
- In step 2 of handling the first 200 “OK” response the a=setup attribute in the SDP will be set to the value of “passive”.
- Before executing step 3 of handling the first 200 “OK” response, the controlling function will start listening for an incoming MSRP session from the inviting CPM group member.
- In step 2, for RCS it is mandatory that Controlling Function buffers the messages exchanged between the time the INVITE was sent and until participants joins in.

As a clarification for RCS:

- In step 4 a, for RCS anonymity will never be allowed
- As RCS does not support pre-defined groups, there will never be an active CPM group session yet
- In case all invited group members sent a SIP final response other than 2xx, the response sent to the inviting CPM group member will be a SIP 480 response

### 9.2.2 Session Cancellation Request

Following differences with [CPMCONVFUNC]:

A CANCEL request will be treated as the CANCEL request in section 9.1.2 of [CPMCONVFUNC] with the clarifications given in this document for section 9.1.2 (i.e., in section 9.1.2 Step 2 when receiving a CANCEL request, the reference to a pre-defined group members is not applicable to RCS, but the same handling will be applied to all members of an ad-hoc group).



Therefore, the CANCEL request shall be rejected with a SIP 481 error response only if a SIP final response was sent to the inviting CPM group member already.

### 9.2.3 Participant Joining a CPM Group Session for a CPM Pre-defined Group

Not applicable for RCS

### 9.2.4 Rejoining CPM Group Session Request

Following differences with [CPMCONVFUNC]:

- In step 3, rather than checking the CPM group Rules which are not applicable for RCS, the CPM Controlling function will check:
  - whether one of the authenticated originator's CPM Addresses of the joining CPM user was part of the list in the invitation for the current session
  - whether one of the authenticated originator's CPM Addresses of the joining CPM user was added as a participant during the course of the session.
  - if configured to do so by Service Provider policy, whether the joining CPM user had not previously explicitly left the CPM Session by sending a SIP BYE request.

Then, if the joining CPM user was not part of the invitation for the current session or had not been invited to be added to the current session via a successful SIP REFER request, or had explicitly left the CPM Session, the request will be rejected in the same way as when the group rules didn't allow the user to join.

- In Step 5, the case for a pre-defined group isn't applicable for RCS, but the CPM Controlling Function will check the operator policy for the maximum allowed number of participants in an ad-hoc group. If that limit is already reached, the same error handling (as in case of an exceeding of the limit) of a pre-defined group will be applied.
- In step 6 if anonymity was requested, a SIP 403 "Forbidden" response including a warning header with the warning text set to "119 Anonymity not allowed" will be returned to the INVITE request
- Before executing step 7, also the SDP offer included in the INVITE request will be checked. If it contains parameters that are not acceptable according to the local server policy or none of the content-types included in it the accept-wrapped-types attribute were part of the initial invitation (see step 3 in chapter 9.2.1) the CPM Controlling Function will return a 488 "Not Acceptable Here" response
- Before executing step 7, the CPM Controlling Function will verify that a Session-Expires header is included in the INVITE request with the refresher parameter set to "uac". If this is not the case, the request will be rejected with a 403 "Forbidden" response that will include a Warning header with the warning text set to "122 Function not allowed"
- In step 8 also Require and Supported headers with the value "timer" will be included as well as a Session-Expires header with the refresher parameter set to "uac". In addition, a contact header including the CPM group session identity with the feature tag is focus and the CPM session ICSI will be included as is a User Agent header indicating the CPM release of the CPM Controlling Function.

- In Step 8 the response will include an SDP description based on the one received in the INVITE request where any content-types included in the accept-wrapped-types that were not part of the session (see above) will be removed from the answer. Furthermore the a=setup: attribute will be set to “passive”.
- Before executing step 9, the CPM Controlling Function will start to listen for an incoming MSRP session from the joining user.
- For RCS the CPM Controlling Function shall also check whether the rejoining user is part of the session already or being invited. If either is the case, the Controlling Function shall handle the situation as described in section 3.4.4.1.7.1 of [RCS 5.3].

As a clarification for RCS:

- In support for long-lived group chat, the CPM Controlling Function may save the CPM Group Session Identity as well as the last participant list for a configurable amount of time according to Service Provider policies, after the CPM Ad-hoc Group Chat has been torn down. If a rejoin request is received when there is no ongoing CPM Ad-hoc Group Chat but the CPM Group Session Identity has been saved, the CPM Controlling Function shall handle the Long-lived CPM Group Chat as per section 9.2.4..
- In step 11, it is possible that all the client notifications for the CPM Group Session Participant Information are routed via the CPM Participating Function, depending on policy in the CPM Participating Function on whether to stay in the path for notifications.

### 9.2.5 Adding Participants Request

Following differences with [CPMCONVFUNC]:

- This handling will also apply in case a REFER request is received without a “method” parameter
- Step 2 is not applicable for RCS.
- After executing step 8, the response will be sent according to the rules and procedures of the SIP/IP core

As a clarification for RCS:

- In step 4, for RCS anonymity will never be allowed

### 9.2.6 Removing Participant Request

Not applicable for RCS

### 9.2.7 Session Leaving Request

Following differences with [CPMCONVFUNC]:

- The case of a predefined group in step 6 a is not applicable for RCS

As clarifications for RCS

- If in step 2, there are still media to be sent to that party, those are discarded

### 9.2.8 CPM Group Session Modification

Not applicable for RCS

### 9.2.9 Group Session Ending Request

Following differences with [CPMCONVFUNC]:

- For RCS these variables are not defined in the XDM policy, but rather managed through a local policy of the CPM controlling function.
- The auto-release variable influencing the release policy is not applicable for RCS
- The case for a pre-defined CPM group session in the number-of-remaining-participants is not applicable for RCS.
- In the release policy section 1 (releasing the remaining participants from the session) whether in item a, the departure of the initiator from the session will trigger a release of the remaining participants from the session will, for RCS, be up to local server policy.
- In the release policy section 1 (releasing the remaining participants from the session), item b (originator leaves in case of a predefined group) is not applicable for RCS.
- In the release policy section 1 (releasing the remaining participants from the session) in item c, the case of a pre-defined CPM group session is not applicable for RCS.

### 9.2.10 Create Session with a Participant

Following differences with [CPMCONVFUNC]:

- The handling will be executed also when a request was sent to add participants as described in chapter 9.2.5. In that case the term “incoming CPM Session invitation” mentioned steps 2, 3, 11 and 12 refers to the received REFER request whereas for the received SDP and the NOTE in step 13, the original INVITE request after the processing of step 3 of chapter 9.2.1 is the relevant request
- Step 7 is not applicable for RCS
- In Step 8, the a=setup: attribute in the SDP will be set to the value of “active”
- Step 9 b is not applicable for RCS
- Step 13 is not applicable for RCS. Any Subject header in the received INVITE request is passed transparently into the generated requests
- When an error response is received, an ACK request will be sent to the party that originated it in order to confirm its proper reception
- In step 15 the MAY is a SHALL for RCS. The policy is thus always applicable.
- When a 200 “OK” response is received and an ACK has been received from the originator of the session, an ACK request will be sent to the party that originated the response. After that an MSRP session will be set up according to the SDP received from that party and notifications will be sent as specified in section 9.2.14.3 to notify those users that have subscribed on participant information that a new participant has joined the conversation. After that the session timer handling will be initiated to monitor the new session.
- For RCS, the case where an ACK is received on the INVITE requests send out according to this section is not applicable.

As a clarification for RCS:

- In all cases a Session-Replaces header with the contribution-ID and Conversation-ID of the original session will be included.

- In case this procedure was started by a received INVITE request, the outgoing INVITE request will include a resource-list body containing the addresses of the invited group members. “To”, “CC” and “BCC” qualifiers will not be included.
- For RCS the interworking to the ISF will be done for a group member when one of the following error responses is received on the INVITE request sent in step 7 to that party:
  - 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 Here”
- Error responses from the ISF/IWF will be treated differently from an error response from the CPM Participants. Regardless of the error type, a request will not be routed to the ISF if it has replied with an error already on that same request
- When a 200 OK response is received, a request for anonymity will never be allowed in RCS

### 9.2.11 Participant Removing Request

Following differences with [CPMCONVFUNC]:

- Removal of Participant via REFER does not apply for RCS, however the procedures are followed when executed based on the criteria in section 9.2.9 for Participant removal via sending BYE request with cause=480 or 410.

As a clarification for RCS:

- In step 3, any media which is left to be sent to that user, will be discarded.

### 9.2.12 Media Plane Handling

No differences with [CPMCONVFUNC].

#### 9.2.12.1 Media Plane Handling for MSRP Sessions

Following differences with [CPMCONVFUNC]:

- The establishment of an MSRP session is done according to the descriptions in the previous chapters taking the received SDP information into account.

As a clarification for RCS:

- Step 3 in “Upon receiving an MSRP SEND request” will be executed as soon as the CPIM headers have been received
- If the total message size is larger than the “max-size” of a particular intended recipient(s) of the original message or contains content not supported by that recipient, the CPM Controlling Function shall not send the message to those intended recipient(s).

- The following clarifies existing OMA CPM functionality and are added as steps 4 and 5 respectively in [CPMCONVFUNC] in the steps under “Upon receiving an MSRP SEND request”:
  - SHALL temporarily queue any messages sent from a participant during the conference startup phase until there is a final response from all invited participants;
  - As each 200 OK is received, SHALL send to that participant all the messages temporarily queued up to that point

#### **9.2.12.2 Media Plane Handling for RTP Sessions**

Not applicable for RCS

#### **9.2.13 Pseudonyms in a CPM Group Session**

As anonymity is not allowed, for RCS only the fact that the CPM Controlling Function should maintain information about the Pseudonyms used in the session is applicable.

#### **9.2.14 Participant Information**

##### **9.2.14.1 CPM Group Session Participant Information Request**

Following differences with [CPMCONVFUNC]:

- If the SUBSCRIBE request does not contain an event header set to “conference” the request will be rejected with a SIP 489 “Bad Event” response that will include a Warning header with the warning text set to “122 Function not allowed”
- In step 1, if the Request URI does not contain a CPM group session identity the request will be rejected with SIP 404 “Not Found” response
- In step 2 is not applicable for RCS
- Step 4 is not applicable for RCS
- In step 6, the contact header will be set to the CPM group session identity
- Step 9 is not applicable for RCS as the Notify is already sent in chapter 9.2.14.3
- For RCS the Controlling Function may also check whether there is already an active subscription for the subscribing participant. If that is the case, the CPM Controlling Function may terminate that earlier subscription before accepting the new one.

As a clarification for RCS:

- Some throttling controlled by local server policy should be applied to the notifications sent during the initial and final phases of the group session

##### **9.2.14.2 Long-lived CPM Group Session**

No differences with [CPMCONVFUNC].

As a clarification for RCS service:

- The duration for which the information is kept is minimally a month.

##### **9.2.14.3 Sending Participant Information Notifications**

Following differences with [CPMCONVFUNC]:

- In step 2 b, the <user> element will be a sub-element of a <users> element

- In step 2 b i the case for privacy which was requested in the first 3 bullets is not applicable for RCS
- In step 2 b l the case for a pre-defined group in the 3rd bullet is not applicable for RCS
- In step 2 b i the 4th bullet is not applicable for RCS

As a clarification for RCS:

- In step 2 b iii, the value provided in the <entity> attribute in the first bullet will be the same address as the one provided in the entity attribute of the <user> element determined in step 2 b i

#### **9.2.14.4 Terminating the Subscription**

Following differences with [CPMCONVFUNC]:

- In step 3 a, the notification shall be generated as described in chapter 9.2.14.3 of [CPMCONVFUNC] and this document with the exception of the final step. That is sending it to the network.

As a clarification for RCS:

- In step 2, the subscription will always be terminated as described in chapter 9.2.7 and chapter 9.2.11 of [CPMCONVFUNC] and this document

#### **9.2.15 Disposition Notification**

No differences with [CPMCONVFUNC].

### **9.3 CPM File Transfer Handling**

No differences with [CPMCONVFUNC].

As a clarification for RCS: the Controlling Function shall take into account the differences and clarifications provided in [RCS 5.3].

#### **9.3.1 CPM File Transfer in Group Session Handling**

No differences with [CPMCONVFUNC].

## **APPENDIX A. Change History**

Appendix not relevant for RCS: as with the other RCS documents the history table is at the end of the document.

## **APPENDIX B. Static Conformance Requirements**

Appendix not relevant for RCS

## **APPENDIX C. CPM-Defined Sip Header Fields**

### **C.1. Header Field Definitions**

#### **C.1.1. Conversation-ID**

No differences with [CPMCONVFUNC].

**C.1.2. Contribution-ID**

No differences with [CPMCONVFUNC].

**C.1.3. InReplyTo-Contribution-ID**

No differences with [CPMCONVFUNC].

**C.1.4. Session-Replaces**

No differences with [CPMCONVFUNC].

**C.1.5. Message-Expires**

No differences with [CPMCONVFUNC].

**C.1.6. Message-UID**

No differences with [CPMCONVFUNC].

**C.1.7. Contact Header field values**

Following differences with [CPMCONVFUNC]:

- For RCS the tag shall be isfocus instead of isFocus.

**C.2. ABNF for the CPM-defined SIP Headers**

No differences with [CPMCONVFUNC].

**C.3. ABNF for the CPM extensions to SIP Headers**

No differences with [CPMCONVFUNC].

**APPENDIX D. Release Version in User-Agent and Server Headers  
(Normative)**

No differences with [CPMCONVFUNC].

**D.1. CPM Version 1.0**

Not applicable for RCS

**D.2. CPM Version 2.0**

No differences with [CPMCONVFUNC].

**APPENDIX E. Examples of CPM-Based Services**

Appendix not relevant for RCS

**APPENDIX F. The Parameters to be Provisioned for CPM Service**

Appendix is not relevant for RCS. Provisioning parameters are described in [RCS 5.3].

**APPENDIX G. Interoperability With OMA Simple IM Clients**

Following differences with [CPMCONVFUNC]:

- The participating function will determine whether a request has to be delivered to a SIMPLE IM client based on the registration information received according to chapter 8.1 of [CPMCONVFUNC] and this document. If there is a SIMPLE IM client registered, the CPM Participating function will deliver the applicable request towards it.
- In the first bullet: for RCS the CPM Participating function shall never have to deliver a CPM Large Message Mode CPM Standalone Message to a SIMPLE IM client. The reference to the +g.oma.sip-im.large-message feature tag is thus not applicable for RCS.
- In the first bullet, if a CPM session or File Transfer needs to be delivered to a SIMPLE IM client, the values for the equivalent feature tags shown in table 2 of [CPMCONVFUNC] will be included in the outgoing INVITE request described in chapter 8.3.2.1 and 8.3.3.1 respectively. If no CPM client needs to be addressed, the CPM feature tags will not be included in the Contact and the Accept-Contact header. If there are also CPM clients to address, the INVITE request will contain both the SIMPLE IM and the CPM feature tags because standard IMS forking will be used. If forking to a specific type of client is desired (e.g. just CPM, or just SIMPLE IM) then the section 6.1.2 applies.
- In the second bullet also a User-Agent header will be added.
- In Note 2, the case for IM Messages, CPM Standalone Messages and Standalone Message disposition notifications is not applicable for RCS
- Note 3 is not applicable for RCS. A SIMPLE IM network based on RCS Release 1-3 is not able to deal with CPM requests. Towards SIMPLE IM networks the NNI for session mode and file transfer will be purely SIMPLE IM based, towards CPM based networks a CPM NNI will be used. How the network comes to the correct NNI type selection and handles a SIMPLE IM NNI in combination with CPM servers is out of scope of RCS. For large message mode and pager mode either an Short Message and Multimedia Messaging Services (SMS/MMS) NNI or a CPM based NNI is used depending on whether or not the terminating network is a CPM based network and a Service Level Agreement (SLA) covering the use of CPM between the networks is in place.
- In the first row of TABLE 4, the case for SIMPLE IM pager mode mapping to CPM is not applicable for RCS.
- The second row of Table 2 (large message mode) is not applicable for RCS

As a clarification for RCS:

- The CPM participating function will offer interoperability with SIMPLE IM on those networks that offer backwards compatibility to clients prior to Release 4.

## **G.1. Media handling with SIMPLE IM Clients**

No differences with [CPMCONVFUNC].

## **G.2. Handling Message Store**

No differences with [CPMCONVFUNC].



## **APPENDIX H. CPM Feature Tags**

No differences with [CPMCONVFUNC]

### **H.1. CPM Feature Identifiers**

No differences with [CPMCONVFUNC].

### **H.2. CPM Client Behaviour**

No differences with [CPMCONVFUNC].

### **H.3. Proposed Formats for CPM Feature Identifiers**

No differences with [CPMCONVFUNC].

### **H.4. Client CPM-based Service Identification**

Not applicable for RCS.

### **H.5. P-Asserted-Service Header Field**

No differences with [CPMCONVFUNC].

## **APPENDIX I. CPM Notification Formats**

### **I.1. Deferred CPM Message Notification Format**

#### **I.1.1. Out-of-band Deferred CPM Message Notification Format**

Not applicable for RCS.

#### **I.1.2. In-band Deferred CPM Message Notification Format**

No differences with [CPMCONVFUNC].

### **I.2. Notification Format of Deferred CPM Message After Expiry**

Not applicable for RCS.

## **APPENDIX J. Deferred Messages Event Package Definition**

No differences with [CPMCONVFUNC].

### **J.1. Deferred Messages Metadata**

#### **J.1.1. Structure**

Following differences with [CPMCONVFUNC]:

- The case for pre-defined groups in the <recipient-list> element is not applicable for RCS

As a clarification for RCS:

- Item 1 for the <message-list> element will always be included, even if set to 0
- Item 3 for the <message-list> element is not applicable for RCS
- Item 1 for the <message > element will always be included

- Item 2 for the <message > element will always be included as this is only used for in-band requests
- Item 3 for the <message > element will always be included for a newly stored message in an in-band notification and for all messages in a notification due to a Deferred CPM Message Information request
- Item 4 for the <message > element will always be included for a newly stored message in an in-band notification and for all messages in a notification due to a Deferred CPM Message Information request
- Item 5 for the <message > element will always be included when present for a newly stored message in an in-band notification and for all messages in a notification due to a Deferred CPM Message Information request
- Item 6 for the <message > element will never be included
- Item 7 for the <message > element will always be included for a newly stored message in an in-band notification and for all messages in a notification due to a Deferred CPM Message Information request
- Item 7 b for the <message > element will always be included when item 7 is present
- Item 7 c for the <message > element will always be included when item 7 is present
- Item 7 d for the <message > element will never be included for RCS
- Item 8 for the <message > element will never be included for RCS
- Item 2 for the <recipient-list> element will never be included for RCS

## **APPENDIX K. Format Of Notification For Deferred CPM Message After Expiry**

Appendix not relevant for RCS

## **APPENDIX L. Emoticons (Normative)**

### **L.1. List of supported emoticons**

No differences with [CPMCONVFUNC].

### **L.2. Emoticon processing by the CPM Clients**

No differences with [CPMCONVFUNC].

#### **L.2.1.Emoticons in composed/sent instant messages**

No differences with [CPMCONVFUNC].

#### **L.2.2.Emoticons in received/displayed instant messages**

No differences with [CPMCONVFUNC].

## Document Management

### Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
1.0	13 Aug 2012	First version for RCS 5.1 Based on the document for RCS 5.0 Approved by DAG and PSMC	PSMC	Tom Van Pelt / GSMA
1.0	26 Sep 2012	Added RCC.11 number		Tom Van Pelt / GSMA
1.0	18 Sep 2013	Converted to Infocentre2 PRD template		Tom Van Pelt / GSMA
2.0	25 Sep 2013	RCS 5.1 Maintenance#2: Include agreed CR1001	RCC TF	Tom Van Pelt / GSMA
3.0	07 May 2014	First version of the document for RCS 5.2: Include approved CR1002	GSG	Tom Van Pelt / GSMA
4.0	28 Feb 2015	First version of the document for RCS 5.3: Include approved CR1003	PSMC	Tom Van Pelt / GSMA

### Other Information

Type	Description
Document Owner	Network 2020 Programme, Global Specification Group
Editor / Company	Tom Van Pelt, GSM Association

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](mailto:prd@gsma.com)

Your comments or suggestions & questions are always welcome.