CPM Conversation Functions

Candidate Version 2.2 – 26 Sep 2017

Open Mobile Alliance

OMA-TS-CPM_Conversation_Function-V2_2-20170926-C

Contents

Figures

Tables

1. Scope

This document provides the technical specifications for the conversation functionality of the CPM Enabler. The document covers the formats and procedures that Principals can use to exchange CPM Standalone Messages, CPM File Transfers and CPM Sessions. Also, it specifies how the CPM Standalone Messages, CPM File Transfers and CPM Sessions are linked to each other in CPM Conversations. The technical specifications are designed to fulfil the requirements, architecture and system concepts that are described in [OMA-CPM-RD], [OMA-CPM-AD] and [OMA-CPM-SD] respectively.

As such, these technical specifications provide the formal definitions of the CPM-PF1, CPM-PF2, and CPM-CF interfaces that have been identified in [OMA-CPM-AD]. Also, these technical specifications formally define the expected behaviour of the CPM Client, CPM Participating Function and CPM Controlling Function functional components that have been identified in [OMA-CPM-AD].

2. References

2.1 Normative References

[3GPP TS 24.229] “IP Multimedia Call Control Protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP)”, 3GPP, TS24.229, URL:http://www.3gpp.org/

[3GPP TS 26.141] “IP Multimedia System (IMS) Messaging and Presence; Media formats and codecs”, 3GPP, TS26.141, URL:http://www.3gpp.org/

[3GPP2 X.S0013.004] “All-IP Core Network Multimedia Domain: IP Multimedia Call Control Protocol Based on SIP and SDP”, 3GPP2, X.S0013.004, URL:http://www.3gpp2.org/

[GSMA IR 84] “Video Share Phase 2 Interoperability Specification, 1.0”, GSM Association, PRD IR.84 1.0, URL:http://www.gsmworld.com/documents/IR.84v1_0.pdf

[GSMA IR.74] “Video Share Interoperability Specification, 1.3”, GSM Association, PRD IR.74 1.3, URL:http://www.gsmworld.com/documents/IR7413.pdf

[GSMA IR.92] “IMS Profile for Voice and SMS, 1.0”, GSM Association, PRD IR.92 1.0, URL:http://www.gsmworld.com/documents/IR9210.pdf

[OMA Client Provisioning] “OMA Client Provisioning”, Open Mobile Alliance™, OMA-ERP-ClientProvisioning-V1_1, URL:http://www.openmobilealliance.org/

[OMA-CPM-AD] “Converged IP Messaging Architecture”, Open Mobile Alliance™, OMA-AD-CPM-V2_2, URL:http://www.openmobilealliance.org/

[OMA-CPM-RD] “Converged IP Messaging Requirements”, Open Mobile Alliance™, OMA-RD-CPM-V2_2, URL:http://www.openmobilealliance.org/

[OMA-CPM-SD] “Converged IP Messaging System Description”, Open Mobile Alliance™, OMA-TS-CPM_System_Description-V2_2, URL:http://www.openmobilealliance.org/

[OMA-CPM_TS_IWF] “CPM Interworking Function; Open Mobile Alliance ™, OMA-TS-CPM_Interworking_Function-V2_1, URL:http://www.openmobilealliance.org/

[OMA-CPM_TS_MessageStorage] “CPM Message Storage; Open Mobile Alliance ™, OMA-TS-CPM_MessageStorage -V2_2, URL:http://www.openmobilealliance.org/

[OMA DM] “OMA Device Management Protocol”, Open Mobile Alliance™, OMA-TS-DM_Protocol-V1_3, URL:http://www.openmobilealliance.org/

[OMA Provisioning Content] “Provisioning Content”, Open Mobile Alliance™, OMA-WAP-TS-ProvCont-V1_1 URL:http://www.openmobilealliance.org/

[OMA-PAP] “Push Access Control”, Open Mobile Alliance™, OMA-TS-PAP-V2_3, URL:http://www.openmobilealliance.org/

[OMA-POC] “Push to talk Over Cellular”, Open Mobile Alliance™, OMA-TS-PoC_System_Description-V2_1, URL:http://www.openmobilealliance.org/

[OMA-Push-Message] “Push Message”, Open Mobile Alliance™, OMA-TS-Push_Message-V2_3, URL:http://www.openmobilealliance.org/

[OMA-Push-OTA] “Push over the Air”, Open Mobile Alliance™, OMA-TS-PushOTA-V2_3, URL:http://www.openmobilealliance.org/

[OMA-SCRRULES] “SCR Rules and Procedures”, Open Mobile Alliance™, OMA-ORG-SCR_Rules_and_Procedures, URL:http://www.openmobilealliance.org/

[OMA-SIMPLE-IM] “Instant Messaging using SIMPLE”; Open Mobile Alliance™, OMA-TS-SIMPLE_IM-V2_0, URL:http://www.openmobilealliance.org/

[OMA-XDM-AD] “XML Document Management Architecture”; Open Mobile Alliance™, OMA-AD-XDM-V2_1, URL:http://www.openmobilealliance.org/

[OMA-XDM-Core] “XML Document Management (XDM) Specification”; Open Mobile Alliance™, OMA-TS-XDM_Core-V2_1, URL:http://www.openmobilealliance.org/

[OMA-XDM-Group] “Group XDM Specification”, Open Mobile Alliance™, OMA-TS-XDM_ Group-V1_1, URL:http://www.openmobilealliance.org/

[OMA-XDM-List] “XML List Specification”; Open Mobile Alliance™, OMA-TS-XDM_List-V2_1, URL:http://www.openmobilealliance.org/

[OMA-XDM-Policy] “Policy XDM Specification”; Open Mobile Alliance™, OMA-TS-XDM _Policy-V1_1, URL:http://www.openmobilealliance.org/

[OMA-XDM-UPP] “UPP Directory XDM Specification”; Open Mobile Alliance™, OMA-TS-XDM_UPP_Directory-V1_0, URL:http://www.openmobilealliance.org/

[RFC2045] "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies, November 1996, URL:http://www.ietf.org/rfc/rfc2045.txt

[RFC2046] “Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types”, N. Freed et al, November 1996, URL:http://www.ietf.org/rfc/rfc2046.txt

[RFC2119] “Key words for use in RFCs to Indicate Requirement Levels”, S. Bradner, March 1997, URL:http://www.ietf.org/rfc/rfc2119.txt

[RFC2183] “Communicating Presentation Information in Internet Messages: The Content-Disposition Header Field”, R. Troost et al, August 1997, URL:http://www.ietf.org/rfc/rfc2183.txt

[RFC2822] “Internet Message Format”, P. Resnick, April 2001, URL:http://www.ietf.org/rfc/rfc2822.txt

[RFC3261] “SIP: Session Initiation Protocol”, J. Rosenberg et al, June 2002, URL:http://www.ietf.org/rfc/rfc3261.txt

[RFC3264] “An Offer/Answer Model with the Session Description Protocol (SDP) ”, J. Rosenberg et al, June 2002, URL:http://www.ietf.org/rfc/rfc3264.txt

[RFC3323] “A Privacy Mechanism for the Session Initiation Protocol (SIP)”, J. Peterson, November 2002, URL:http://www.ietf.org/rfc/rfc3323.txt

[RFC3325] “Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks”, C. Jennings et al, November 2002, URL:http://www.ietf.org/rfc/rfc3325.txt

[RFC3326] “The Reason Header Field for the Session Initiation Protocol”, H. Schulzrinne, December 2002, URL:http://www.ietf.org/rfc/rfc3326.txt

[RFC3420] “Internet Media Type message/sipfrag”, R. Sparks, November 2002, URL:http://www.ietf.org/rfc/rfc3420.txt

[RFC3428] “Session Initiation Protocol (SIP) Extension for Instant Messaging”, B. Campbell (Editor), December 2002, URL:http://www.ietf.org/rfc/rfc3428.txt

[RFC3458] “Message Context for Internet Mail”, E. Burger, January 2003, URL:http://tools.ietf.org/rfc/rfc3458.txt

[RFC3501] “Internet Message Access Protocol - Version 4rev1”, M. Crispin, March 2003, URL:http://www.ietf.org/rfc/rfc3501.txt

[RFC3515] “The Session Initiation Protocol (SIP) Refer Method”, R. Sparks, April 2003, URL:http://www.ietf.org/rfc/rfc3515.txt

[RFC3550] “RTP: A Transport Protocol for Real-Time Applications”, H. Schulzrinne et al, July 2003, URL:http://www.ietf.org/rfc/rfc3550.txt

[RFC3680] “A Session Initiation Protocol (SIP) Event Package for Registrations”, J. Rosenberg, March 2004, URL:http://www.ietf.org/rfc/rfc3680.txt

[RFC3840] “Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)”, J. Rosenberg et al, August 2004, URL:http://www.ietf.org/rfc/rfc3840.txt

[RFC3841] “Caller Preferences for the Session Initiation Protocol (SIP)”, J. Rosenberg et al, August 2004, URL:http://www.ietf.org/rfc/rfc3841.txt

[RFC3862] "Common Presence and Instant Messaging (CPIM): Message Format", G. Klyne et al, August 2004, URL:http://www.ietf.org/rfc/rfc3862.txt

[RFC3892] "The Session Initiation Protocol (SIP) Referred-By Mechanism", R. Sparks, September 2004, URL:http://www.tools.ietf.org/rfc/rfc3892.txt

[RFC3903] "Session Initiation Protocol (SIP) Extension for Event State Publication", A. Niemi (Editor), October 2004, URL:http://www.ietf.org/rfc/rfc3903.txt

[RFC3966] "The tel URI for Telephone Numbers" IETF RFC, H. Schulzrinne, December 2004

URL:http://tools.ietf.org/html/rfc3966

[RFC3994] Indication of Message Composition for Instant Messaging IETF RFC

URL:http://tools.ietf.org/html/rfc3994

[RFC4028] "Session Timers in the Session Initiation Protocol (SIP)", S. Donovan et al, April 2005, URL:http://www.ietf.org/rfc/rfc4028.txt

[RFC4122] "A Universally Unique IDentifier (UUID) URN Namespace", P. Leach et al, July 2005, URL:http://www.ietf.org/rfc/rfc4122.txt

[RFC4353] "Conferencing Framework with SIP", J. Rosenberg, February 2006, URL:http://www.ietf.org/rfc/rfc4353.txt

[RFC4483] "A Mechanism for Content Indirection in Session Initiation Protocol (SIP) Messages", E. Burger (Editor), May 2006, URL:http://www.ietf.org/rfc/rfc4483.txt

[RFC4488] "Suppression of Session Initiation Protocol (SIP) REFER Method Implicit Subscription", O. Levin, May 2006, URL:http://www.ietf.org/rfc/rfc4488.txt

[RFC4566] "SDP: Session Description Protocol", M. Handley et al, July 2006, URL:http://www.ietf.org/rfc/rfc4566.txt

[RFC4575] "A Session Initiation Protocol (SIP) Event Package for Conference State", J. Rosenberg et al, August 2006, URL:http://www.ietf.org/rfc/rfc4575.txt

[RFC4579] "Session Initiation Protocol (SIP) Call Control - Conferencing for User Agents", A. Johnston et al, August 2006, URL:http://www.ietf.org/rfc/rfc4579.txt

[RFC4975] "The Message Session Relay Protocol (MSRP)", B. Campbell (Editor), September 2006, URL:http://www.ietf.org/rfc/rfc4975.txt

[RFC5092] "IMAP URL Scheme", A. Melnikov (Editor), November 2007, URL:http://www.ietf.org/rfc/rfc5092.txt

[RFC5318] "The Session Initiation Protocol (SIP) P-Refused-URI-List Private-Header (P-Header) ", J. Hautakorpi et al, December 2008, URL:http://www.ietf.org/rfc/rfc5318.txt

[RFC5364] "Extensible Markup Language (XML) Format Extension for Representing Copy Control Attributes in Resource Lists", M. Garcia-Martin et al, October 2008, URL:http://www.ietf.org/rfc/rfc5364.txt

[RFC5365] "Multiple-Recipient MESSAGE Requests in the Session Initiation Protocol", M. Garcia-Martin et al, October 2008, URL:http://www.ietf.org/rfc/rfc5365.txt

[RFC5366] "Conference Establishment Using Request-Contained Lists in the Session Initiation Protocol (SIP)", G. Camarillo et al, October 2008, URL:http://www.ietf.org/rfc/rfc5366.txt

[RFC5368] "Referring to Multiple Resources in the Session Initiation Protocol (SIP)", G. Camarillo et al, October 2008, URL:http://www.ietf.org/rfc/rfc5368.txt

[RFC5438] "Instant Message Disposition Notification (IMDN)", E. Burger et al, February, 2009, URL:http://tools.ietf.org/html/rfc5438

[RFC5438Errata] Instant Message Disposition Notification (IMDN) IETF RFC 5438 Errata ID 3013

URL:http://www.rfc-editor.org/errata_search.php?rfc=5438 (see also section C.2)

[RFC5547] "A Session Description Protocol (SDP) Offer/Answer Mechanism to Enable File Transfer", M. Garcia-Martin et al, May 2009, URL:http://www.ietf.org/rfc/rfc5547.txt

[RFC5626] "Managing Client-Initiated Connections in the Session Initiation Protocol (SIP)", C. Jennings (Editor), October 2009, URL:http://www.ietf.org/rfc/rfc5626.txt

[RFC5627] “Obtaining and Using Globally Routable User Agent URIs (GRUUs) in the Session Initiation Protocol (SIP)”, J. Rosenberg, October 2009, URL:http://www.ietf.org/rfc/rfc5627.txt

[RFC5628] “Registration Event Package Extension for Session Initiation Protocol (SIP Globally Routable User Agent URIs (GRUUs)”, P. Kyzivat, October 2009, URL:http://www.ietf.org/rfc/rfc5628.txt

[RFC5806] “Diversion Indication in SIP”, S. Levy, March 2010, URL:http://www.ietf.org/rfc/rfc5806.txt

[RFC5876] “Updates to Asserted Identity in the Session Initiation Protocol (SIP)”, J. Elwell, April 2010, URL:http://www.ietf.org/rfc/rfc5876.txt

[RFC6135] “An Alternative Connection Model for the Message Session Relay Protocol”, C. Holmberg et al, June 2010, URL:http://www.ietf.org/rfc/rfc6135.txt

[RFC6665] “SIP-Specific Event Notification”, A. B. Roach, July 2012, URL:http://www.ietf.org/rfc/rfc6665.txt

[RFC6714] “Connection Establishment for Media Anchoring (CEMA) for the Message Session Relay Protocol (MSRP)”, C. Holmberg et al, August 2012, URL:http://www.ietf.org/rfc/rfc6714.txt

[RFC7647] “Clarifications for the Use of REFER with RFC6665”, R. Sparks et al, September 2015, URL:https://tools.ietf.org/html/rfc7647

[RFC7701] “Multi-party Chat Using the Message Session Relay Protocol (MSRP)”, M. Garcia-Martin et al, December 2015, URL:http://www.ietf.org/rfc/rfc7701.txt

2.2 Informative References

[OMA EVC] “OMA Enhanced Visual Call V1.0”, Open Mobile Alliance™, URL:http://www.openmobilealliance.org/

[OMA PDE] “OMA Presence SIMPLE Data Extensions V1.3” , Open Mobile Alliance™, URL:http://www.openmobilealliance.org/

[OMA Pres] “OMA Presence SIMPLE”, Version 2.0, Open Mobile Alliance™, URL:http://www.openmobilealliance.org/

[OMADICT] “Dictionary for OMA Specifications”, Open Mobile Alliance™, OMA-ORG-Dictionary-V2_9, URL:http://www.openmobilealliance.org/

3. Terminology and Conventions

3.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope” and “Introduction”, are normative, unless they are explicitly indicated to be informative.

3.2 Definitions

Blacklist	A list of users that are blocked from performing a CPM functionality.
Conference Information Object	See [OMA-CPM-SD].
CPM 1-1 Session	See [OMA-CPM-SD].
CPM Address	See [OMA-CPM-RD].
CPM Ad-hoc Group	See [OMA-CPM-RD].
CPM Chat Message	See [OMA-CPM-RD].
CPM Client	See [OMA-CPM-SD].
CPM Closed Group Session	See [OMA-CPM-RD].
CPM Contribution Identity	See [OMA-CPM-SD].
CPM Controlling Function	See [OMA-CPM-AD].
CPM Conversation	See [OMA-CPM-RD].
CPM Conversation History	See [OMA-CPM-RD].
CPM Conversation Identity	See [OMA-CPM-SD].
CPM Feature	A basic feature offered by the CPM Enabler (e.g. Pager Mode messaging, Large Message Mode messaging, session-based messaging, file transfer, deferred messaging, etc).
CPM Feature Tag	See [OMA-CPM-SD]
CPM File Transfer	See [OMA-CPM-SD].
CPM Group Session	See [OMA-CPM-RD].
CPM Group Session Identity	See [OMA-CPM-SD].
CPM Long-lived Session	See [OMA-CPM-RD].
CPM Message	See [OMA-CPM-RD].
CPM Participating Function	See [OMA-CPM-AD].
CPM Pre-defined Group	See [OMA-CPM-RD].
CPM Session	See [OMA-CPM-RD].
CPM Session Invitation	See [OMA-CPM-RD].
CPM User	See [OMA-CPM-RD].
CPM User Activity	See [OMA-CPM-AD].
CPM-based Service	See [OMA-CPM-AD].
Deferred CPM Message	See [OMA-CPM-SD].
Device	See [OMADICT].
Enabler	See [OMADICT].
Group State Object	See [OMA-CPM-SD]

Group XDMS	See [OMA-XDM-AD].
Interworking Function	See [OMA-CPM-AD].
Interworking Selection Function	See [OMA-CPM-AD].
Join-in Group	See [OMA-XDM-Group].
Large Message Mode	See [OMA-CPM-AD]
Large Message Mode	See [OMA-CPM-AD].
Media	See [OMA-CPM-RD].
Media Object	See [OMA-CPM-AD].
Media Plane	See [OMA-CPM-AD].
Media Stream	As defined in [RFC3264].
Message Storage Server	See [OMA-CPM-AD].
Pager Mode	See [OMA-CPM-AD]
Participant	See [OMADICT].
Participant Information	See [OMA-CPM-SD].
Primary Device	See [OMA-CPM-RD].
Primary Identity	See [OMA-CPM-RD].
Principal	See [OMADICT].
Pseudonym	See [OMA-CPM-RD].
Registration Event Information	See [OMA-CPM-SD].
Secondary Device	See [OMA-CPM-RD].
Session Info Object	A MIME object stored in the CPM Message Store Server, that persists the CPM Session data received in the SIP INVITE request, e.g. the SIP headers and the SIP body.
Signalling Path	See [OMA-CPM-SD].
Unique User Agent Identifier	See [OMA-CPM-SD].
User Preferences Profile	See [OMA-CPM-RD].

3.3 Abbreviations

3GPP	See [OMADICT]
3GPP2	See [OMADICT]
ABNF	See [OMADICT]
B2BUA	See [OMADICT]
CPIM	Common Presence and Instant Messaging
CPM	See [OMADICT]
EVC	Enhanced Visual Call
GRUU	Globally Routable User Agent URI
IETF	See [OMADICT]
IMAP	See [OMADICT]
IMDN	See [OMADICT]
IMS	See [OMADICT]
IP	See [OMADICT]
ISF	See [OMADICT]

IWF	See [OMADICT]
MIME	See [OMADICT]
MSRP	See [OMADICT]
OMA	See [OMADICT]
OMNA	Open Mobile Naming Authority
RFC	See [OMADICT]
RTCP	RTP Control Protocol
RTP	Real-time Transport Protocol
SCR	Static Conformance Requirements
SDP	See [OMADICT]
SIMPLE	See [OMADICT]
SIP	See [OMADICT]
TCP	See [OMADICT]
UA	User Agent
UAC	User Agent Client
UAS	User Agent Server
UDP	User Datagram Protocol
URI	See [OMADICT]
URL	See [OMADICT]
WAP	See [OMADICT]
XDM	See [OMADICT]
XDMS	See [OMADICT]
XML	See [OMADICT]

Note: Abbreviations defined in the OMA Dictionary complements this section.

4. Introduction

The Converged IP Messaging (CPM) Enabler provides the convergence of multimedia communication services which accommodate different user experiences such as deferred and immediate messaging, session-based messaging, and half duplex/full duplex conferencing, while leveraging standardized service functionalities from existing communication Enablers like Instant Messaging [OMA-SIMPLE-IM] or Push to talk over Cellular [OMA-POC].

The CPM Enabler provides a framework by defining a horizontal Enabler built on top of a SIP/IP core infrastructure. This framework comprises a set of functional components and interfaces that have been designed to facilitate easy deployment of existing and future communication services. The components of the CPM Enabler are independently reusable. The set of functions interact with one another via the framework provided by the CPM Enabler. The CPM Enabler offers multimedia communication functionalities and can be used to build Services, a.k.a. CPM-based Services. An instantiation of a CPM communication can have one or more media types.

All CPM-based Services will use the functional components and interfaces provided by the framework. Non-CPM Communication Services require Interworking Functions that provide adaptation/mapping between the CPM Enabler and the different technologies of these Non-CPM Communication Services to communicate with the framework.

4.1 CPM Version 1.0

CPM Enabler version 1.0 Conversation Function offers:

- Registration
    - handling 3rd party CPM service registrations
    - handling Registration Event Information
- CPM Client function, CPM Participating Function, CPM Controlling Function
- CPM Standalone Message handling
    - sending/receiving Pager Mode CPM Standalone Message, Large Message Mode CPM Standalone Message
    - with/without disposition notifications
    - immediate/deferred delivery of CPM Standalone Message
- CPM File Transfer
    - initiating/receiving/rejecting CPM File Transfer
- supports discrete (e.g. text, image, video clip, audio clip, voice clip, binary file) Media Types and continuous (e.g. bidirectional voice, streaming video) Media Stream Types
- CPM 1-1 Session handling
    - initiating/receiving/modifying/closing CPM 1-1 Session
    - media plane handling
    - extending CPM 1-1 Session to CPM Group Session
- CPM Group Session handling
    - initiating/receiving/modifying/closing CPM Group Session for CPM Ad-hoc Groups and CPM Pre-defined Groups
    - media plane handling
    - adding/removing users at any time during a CPM Session
    - receiving participant information
- supports multiple device environment
- applying appropriate User Preferences Profiles among several User Preferences Profiles like Office, Home, Meeting, etc. to which the preferences of the CPM User are associated for message delivery

- supports if interworking decision with Non-CPM Communication Services is required or not
- recording of CPM Standalone Messages, CPM File Transfers and CPM Sessions, including any Media Objects attached to them in a network-based storage subject to the preferences of the CPM User and service provider policies

4.2 CPM Version 2.0

CPM Enabler version 2.0 Conversation Function adds the following enhancements:

- CPM Standalone Message handling:
  - Enhanced multi-device functionality for sending imdn notifications (delivery notifications and read reports) towards the device that originated the CPM Standalone Message;
  - Delivery to all suitable registered CPM Clients.
- CPM 1-1 and Group Session handling:
  - Support for notifications within the chat such as “isComposing” and disposition notifications (i.e. delivered, displayed);
  - Storing of chat messages and associated notifications while a chat participant is not available (e.g. lost coverage, not registered in IMS) or missed chat;
  - Delivery of stored chat messages and disposition notifications once the recipient becomes available;
  - Enhanced multi-device functionality for chat (forking in case of auto-answer):
    - Delivery to all suitable registered CPM Clients, but only the CPM Client on which the user has composed a message is kept in the session.
  - Support for live recording of chat messages;
  - Support Closed Group Session;
  - Support Long-lived Group Session;
  - Interworking over the NNI with SIMPLE IM V2.0 chat.
- File transfer:
  - Support for file size, file name, and type in a CPM File Transfer request;
  - Define a maximum file size policy;
  - Support CPM File Transfer while having an ongoing CPM Session;
  - Support CPM File Transfer to off-line recipients;
  - Support File Transfer resume by either sender or recipient;
  - Support thumbnail in the invitation of the content to be transferred;
  - Deferral of received associated IMDN notifications for a previously originated CPM File Transfer , while the CPM User is not available;
  - Request sent to all suitable registered CPM Clients and delivered to the first device that answers.
- Addition of sip.instance mechanism for multi-device support, for:
  - delivery of disposition notifications to the CPM User device that has originated the message for which the associated IMDN is being delivered, and
  - deferred delivery of chat, group chat messages, File Transfer and associated disposition notifications to the same CPM User’s device on which the invitation was originally accepted, before being involuntarily disconnected during the SIP session.

  Backward compatibility with version 1.0:

- The MSRP session matching and connection method adopted in version 2.0 is based on the [RFC6714] which is incompatible with the CPM version 1.0 MSRP session matching based on the expired IETF draft “Session Matching Update for the Message Session Relay Protocol (MSRP) version 10, IETF, April 2010”.
- Excludes support of session history object as defined in CPM version 1.0. This has been replaced in CPM version 2.0 by the session history folder and session info object.

4.3 CPM Version 2.1

CPM Enabler version 2.1 Conversation Function provides the following changes and enhancements:

- Handling of IMDN notifications:
  - delivery of disposition notifications to the CPM User that has originated that CPM Message, CPM Chat Message or CPM File Transfer, instead of delivery only to his/her device that has originated that CPM request associated with the disposition notification;
- Capability to negotiate the MSRP chunk size to be used in CPM requests, to leverage the various network access types bandwidths and service level agreements between domains.
- Support for additional service provider delivery policies:
  - Immediate Delivery feature:
    - Enables immediate delivery of received CPM Chat Messages, CPM Standalone Messages, CPM File Transfers and disposition notifications to all of the suitable registered recipient’s CPM Clients,
    - Enables immediate delivery of CPM Chat Messages, CPM Standalone Messages, CPM File Transfers and associated disposition notifications that originated on one of a CPM User’s multiple connected CPM Clients, to all the other suitable registered CPM User’s Clients.
- Converged messaging procedures via either SIP/IP Core or interworked delivery via non-CPM technologies, for CPM Standalone Messages, CPM Sessions and CPM File Transfers on any available access domain on which the CPM User is reachable delivery, for the following features:
  - Deferral of CPM Messages, CPM Sessions and CPM File Transfer, and
  - Delivery on any access domain (IP or non-IP) available, per service provider policies, and
  - Common storage functions.
- Event Reporting Framework:
  - Provides a set of procedures with a choice of SIP delivery methods that allow:
    - a client to report events that occured on the device, or
    - a server to report events of interest for another network entity, or
    - for the server to request reporting of these events, or
    - to a application to use CPM Event Reporting Framework as a transport for its own requests.
  - Several CPM defined events are supported such as:
    - IMAP events between CPM Client and the CPM Participating Function;
      - Update of IMAP flags for one, or more, CPM Message Store objects based on CPM User actions that have occurred on the CPM Client;
    - Interworking events reported by the CPM Interworking Function to the CPM Participating Function;
    - Activity events reported by the CPM Client to the CPM Participating Function, or requested by the CPM Participating Function from a CPM Client.

4.4 CPM Version 2.2

CPM Enabler version 2.2 Conversation Function provides the following changes and enhancements:

- Support for OMA NMS APIs as RESTful interface and REST-based data model for storing objects in the CPM Message Store, when recording of CPM Standalone Messages, CPM Sessions, CPM File Transfer History and Standalone Media Objects;
- CPM Group Session:
    - enhancements to Group Session Data Management (GSDM):
        - Added support for the group chat bulletin;
        - Added procedures for the Group Chat administrator to give this role to another user, or to take it away;
        - Add minimum policies for the user admin support (single admin, any admin);
        - Schema updates for icon to support both URL and direct inclusion of the file icon in the CPM Group Session Data Management information;
    - CPM Group communications enhancements (Group Session and Standalone Message):
        - Support for denoted messages in Group communications via CPIM “cc” header;
    - Optimization decisions:
        - When supported, propagating the Participant List updates in MSRP SEND packets in an active CPM Group Session, to avoid establishing separate SIP dialogs for the SIP NOTIFY requests towards each Participant (or towards each Participant’s connected devices, in multi-device direct delivery cases)
            - This optimizes the CPM Participating Function -CPM Client flow, CPM Controlling Function-CPM Participating Function flow;
        - use the conference event package information as basis for storing Participant Information, CPM Groups Session Data Management information and for transmitting these updates in the CPM Group Session (in MSRP);
- for the case of the direct delivery to multiple user devices (registered and connected in a CPM communication): reinstated the use of instance-ID to reach out to specific CPM User device;
- Bug fixes and corrections from previous releases:
    - add missing procedures for handling of direct delivery to multiple CPM User devices when messages originated by the CPM User have to be forked to all his other registered & connected devices to that CPM Session, CPM Standalone Messages Pager Mode and Large Message Mode;
    - Explicitly clarifying in the procedures for all network entities used in a CPM flow, the mandatory preservation in SIP B2BUA requests/responses, of the conference focus uri (tagged with ‘isfocus’) in the Contact header field;
    - Alignments with newly published [RFC7647] for handling of in SIP REFER (Refer-Sub header field by network entities along the path in B2BUA flows);
    - Handling of default cases such as empty subject, empty icon, empty bulletin in the CPM Group Session.

5. Format of CPM Conversation Items

5.1 CPM Standalone Message

A CPM Standalone Message is either a Pager Mode CPM Standalone Message carried in a SIP MESSAGE request as described in [RFC3428] or a Large Message Mode CPM Standalone Message. The size of a Pager Mode CPM Standalone Message should not exceed 1300 bytes. If a CPM Standalone Message is larger than 1300 bytes, the contents of the message are not inserted into the SIP MESSAGE request but are carried via MSRP as defined in [RFC4975] and [RFC6714]. In that case, a SIP session is established between the interested parties (sender and all receivers) with MSRP as the Media Stream. The CPM Standalone Message contents are then transmitted via MSRP, using chunking if necessary. The MSRP chunk size MAY be negotiated between endpoints, to determine the maximum MSRP chunk size to be used in a SIP session, as described in section 5.2.1 “*SDP Contents for CPM Sessions*”. This SIP session should *not* be confused with a CPM Session as no CPM Session is established. The SIP session is only used to transmit exactly one Large Message Mode CPM Standalone Message after which the SIP session is torn down. The CPM Standalone Message contents are contained in a CPIM wrapper as defined in [RFC3862].

5.2 CPM Session

A CPM Session uses SIP session functionality as defined by [RFC3261] to exchange multimedia content as well as CPM Chat Messages between two or more CPM Users or between CPM Users and non-CPM Principals. A CPM Session can be established between two Principals as a CPM 1-1 Session or between several Principals as a CPM Group Session. In the latter case, a CPM Controlling Function acts as a conference focus as defined by [RFC4353].

A CPM 1-1 Session can be extended by one of the Participants to a CPM Group Session as described in section 0 “*Extending a CPM 1-1 Session to a CPM Group Session*”.

A CPM User can invite other CPM and non-CPM Principals to a CPM Session. The invitees are either members of a CPM Pre-defined Group or a number of ad-hoc recipients that the CPM User selects dynamically. A CPM Pre-defined Group is a “Shared Group” as described in [OMA-XDM-Group]. Ad-hoc recipients are carried as a 'recipient-list' body in the SIP INVITE request as described in [RFC5366].

The duration of a CPM Session depends on whether it is a CPM 1-1 Session or a CPM Group Session. In the case of a CPM 1-1 Session, the session is terminated when one of the two Participants leaves the session. In the case of a CPM Group Session for a CPM Ad-hoc Group, the session usually ends when the session initiator ends the session. In the case of a CPM Group Session for a CPM Pre-defined Group, the ending of the session depends on the conditions set in the policy of that group as described in [OMA-XDM-Group].

For Long-lived CPM Group Sessions, the session ends temporarily when the inactivity timer is reached (i.e. no CPM user messages are exchanged for a period of time). An inactive Long-lived CPM Group Session can be re-started by any of the Participants in that CPM Group Session at the time when inactivity ended the session.

Following notifications are provided for CPM Chat Messages:

- “isComposing”,
- delivery notifications,
- read reports notifications, and’
- interworking notifications

to the CPM User’s device that has sent the CPM Chat Message.

The delivery notifications and read reports are stored also as part of the Conversation History that includes the CPM Session for the CPM User that sent the original message.

The delivery notifications and read reports can be sent within the CPM Session, or outside of the CPM session if they are generated after the CPM Session was terminated.

The interworking notifications are received when a CPM Chat Message was delivered via an interworking method.

The “isComposing” notifications are generated during the CPM Session according to the rules and procedures of [RFC3994]. A CPM Client receiving a MSRP SEND message holding an “isComposing” status SHOULD change the “isComposing” status of the sender according to the rules and procedures of [RFC3994].

In a 1-1 CPM Session, “isComposing” status SHALL be sent without CPIM wrapper and therefore no delivery and/or display notification shall be requested.

The originating CPM Participating Function SHALL always set the CPIM “DateTime” header value in the CPM Chat Messages it receives from the served CPM User’s Client. The CPM Participating Function SHALL also set the CPIM “DateTime” header value and IMDN “DateTime” element in notifications. In both cases, the CPM Participating Function SHALL overwrite any “DateTime” information provided by the CPM Client. In case of retransmissions of the CPM Chat Messages or IMDNs, the Originating Participating Function SHOULD overwrite the “DateTime” headers value with the timestamp of the last retransmission. A CPM Client receiving these CPM Chat Messages SHOULD therefore rely on these headers fields to contain the correct time, and can for example use the provided information to ensure correct ordering of the sent and received CPM Chat Messages and CPM notifications (i.e. disposition notifications and/or “isComposing”). The CPM Client SHOULD perform any timezone conversions, if necessary.

Upon a successful delivery to at least one of the CPM User’s Clients, via either CPM Session mechanism or via Interworking to legacy technologies, the delivered CPM Session information is stored in the CPM Message Store by the CPM Participating Function. The stored information includes: the SIP session information, the CPM Chat Messages, and if applicable their IMDN notifications and Group State Object(s) and/or Conference Information Object(s).

When the CPM Session invitation could not be delivered to any of the CPM User’s Clients, the CPM Participating Function MAY defer the CPM Session and its CPM Chat Messages, associated IMDNs, and if applicable any Group State Object(s) information, for delivery to the CPM User’s Client(s) at a later time.

5.2.1 SDP Contents for CPM Sessions

5.2.1.1 SDP Contents when Initiating or Modifying a CPM Session

An initiating entity (e.g. a CPM Client, CPM Participating Function, CPM Controlling Function or an IWF) SHALL populate the SDP of a CPM Session Invitation or a CPM Session modification request to match the Media Streams that are needed to be set up, deleted or modified. Therefore the initiating entity SHALL include in the SIP INVITE request a MIME SDP body as an SDP offer according to the rules and procedures of [RFC4566] and [RFC3264]. The SDP offer SHALL contain media descriptions matching the requested Media Streams according to the following clarifications:

- When including an offer for a Media Stream for CPM Chat Messages, using MSRP, the initiating entity SHALL include a media description according to the rules and procedures of [RFC4975] and [RFC6714].
- When including an offer for a Media Stream for real-time continuous Media, using RTP/RTCP, the initiating entity SHALL include a media description according to the rules and procedures of [RFC3264] and [RFC3550].
- When including an offer for a Media Stream of another Media Stream Type, the initiating entity SHALL follow the respective standard for that Media Stream Type.

In the case of CPM Session modification, the modified SDP SHALL follow the rules and procedures for modifying a session described in [RFC3264].

When the SIP/IP core corresponds with 3GPP/3GPP2 IMS, the SDP offer SHALL also adhere to the rules and procedures described in [3GPP TS 24.229] / [3GPP2 X.S0013.004].

An initiating entity (e.g. a CPM Client or an IWF) SHALL support “isComposing” notifications for CPM Chat Messages and SHALL populate in the SDP of the SIP INVITE request and response the value "application/im-iscomposing+xml" in the a=accept-types attribute as described in [RFC3994] for CPM 1-1 Sessions and in the a=accept-wrapped-types attribute as described in [RFC3994] for CPM Group Sessions.

An initiating entity requesting support for CPM Event Reporting as defined in section 6.7 “*CPM Event Reporting Framework*” SHALL populate in the SDP of the SIP INVITE request and response the value of the CPM Event Reporting Content-Type as defined in section 6.7.4 “*CPM Event Reporting Data Format*” in the *a=accept-wrapped-types* attribute.

An initiating entity (i.e. a CPM Client, CPM Participating Function, CPM Controlling Function or an IWF) SHALL populate the SDP a=setup attribute in SDP offers with the “actpass” value according to [RFC6135]. Upon returning a SIP ACK after receiving an SIP 200 “OK” response:

A. the CPM Client SHALL always act as an active MSRP endpoint and SHALL initiate the MSRP connection according to [RFC6135];

B. all CPM entities except for the CPM Client, SHALL act as negotiated i.e. if the received SDP answer contained an a=setup attribute set to the value:

   i. “active”, then SHALL act as a “passive” endpoint and SHALL start listening for the incoming MSRP session;

   ii. otherwise, SHALL act as an “active” endpoint and SHALL initiate the transport connection for MSRP.

An initiating entity (e.g. a CPM Client, CPM Participating Function or an IWF) SHALL also populate in the SDP of the SIP INVITE request and response the value *"message/imdn+xml"*, in the *a=accept-wrapped-types* attribute as per rules and procedures of [RFC5438] and [RFC5438Errata].When negotiation of the maximum MSRP chunk size is supported, the initiating entity SHALL populate the SDP a-line attribute “a=max-chunk-size” containing a value in Kbytes starting from 100 KB. When in the SDP answer a value is provided, this SHALL be handled as the maximum chunk size for both sending and receiving (i.e. the initiating entity does not have to be prepared to support larger chunks).

For backward compatibility with [RFC4975], an initiating entity that has populated the “a=max-chunk-size” attribute but does not receive the MSRP chunk size attribute in the SDP answer, SHALL assume [RFC4975] behaviour i.e. that there is no limit set to the MSRP chunk size and it SHOULD use its own value (e.g. implementations may provide different configured values per network access type: LTE, Wi-Fi, 3G).

5.2.1.2 SDP Handling at Intermediate Nodes

Intermediate nodes (e.g. a CPM Participating Function, a CPM Controlling Function or an IWF) SHALL include the contents of the SDP they received in the SDP they send out, in accordance with the rules and procedures of [RFC3264]. Specific attributes in the SDP MAY be modified for the following reasons:

- To remove or modify media descriptions that are not allowed according to service provider policies or CPM Group policy or user preferences.
- To modify IP-address and port information in the SDP c and m lines and if it acts as MSRP B2BUA also sets the SDP a=path attribute to insert the intermediate entity in the media path of the session.
- To insert the SDP 'msrp-cema' attribute in SDP offers it sends, and then process the response as per MSRP session matching according to section 5.2.1.4 “*Handling of Media connection parameters for MSRP*”.
- To remove the support of CPM Event Reporting when the intermediate node acts as a MSRP B2BUA and the intended recipient of the SIP request is a different CPM User than the sender. In such case, once the MSRP sessions are established, the intermediate node SHALL filter out the CPM event reports (i.e. the event reports received by the originating CPM Participating Function from a CPM Client SHALL not be propagated to the terminating network).

All modifications SHALL be according to the rules and procedures of [RFC3264] and the respective Media Stream standards (i.e. [RFC4975] and [RFC6714] for MSRP-based media description and [RFC3264] and [RFC3550] for RCP/RTCP-based media descriptions).

The intermediate node that supports MSRP chunk negotiation:

a) SHALL check the SDP a-line attribute “a=max-chunk-size” to determine if the proposed MSRP chunk size is appropriate for both sending and receiving of MSRP streams to and from the entity with which the SDP negotiating is being done based on its own configured service provider policies, as applicable:
   1) for the network access type, or
   2) the service level agreement for inter-domain requests.

   NOTE: intermediate nodes SHOULD ignore and pass transparently all SDP attributes they do not understand.

b) If the value of the MSRP chunk size received in the SDP offer or answer is not acceptable, based on the service provider policies, and if:
   i. the B2BUA intermediate node is capable to re-chunk the MSRP requests, then:
      1) it SHOULD populate the SDP a-line attribute “a=max-chunk-size” with the appropriate configured value, in the SDP offer of the outgoing SIP INVITE request, or SIP response.
   ii. If the B2BUA intermediate node is not capable of using a different MSRP chunk size, such as a specific value provided by the SDP offerer in the SIP request, or by the SDP answerer in the SIP response, or it does not match its own configured values for its current network access type, then:

1) it SHALL use the closest configured MSRP chunk size that is lower than the other endpoint’s value.

c) If the value of the MSRP chunk size received in the SDP offer or answer is acceptable, intermediate node SHALL transparently pass the received value of the “a=max-chunk-size” in the SDP offer or answer.

Otherwise, in all other cases:

– the intermediate node SHOULD transparently pass the received value of the “a=max-chunk-size” in the SDP offer or answer.

An intermediate node (i.e. a CPM Participating Function, a CPM Controlling Function or an IWF) SHALL populate the SDP a=setup attribute in:

A) SDP offers, with the “actpass” value according to [RFC6135]. Upon returning a SIP ACK after receiving an SIP 200 “OK” response, SHALL act as negotiated i.e. if the received SDP answer contained an a=setup attribute set to the value:

1) “active”, then SHALL act as a “passive” endpoint and SHALL start listening for the incoming MSRP session;

2) otherwise, SHALL act as an “active” endpoint and SHALL initiate the transport connection for MSRP.

B) SDP answers, with the “passive” value according to [RFC6135] and SHALL further act as a passive MSRP endpoint according to [RFC6135].

5.2.1.3 SDP Handling at Terminating Nodes

A terminating entity (e.g. a CPM Client or an IWF) SHALL process an incoming SDP and accept, modify or reject the Media Streams requested in the incoming SDP as defined by [RFC3264]. The terminating entity SHALL handle the media descriptions according to the following clarifications:

- Media descriptions for a Media Stream for CPM Chat Messages, using MSRP, SHALL be handled and responded to according to the rules and procedures of [RFC4975] and [RFC6714].
- Media descriptions for a Media Stream for real-time continuous Media, using RTP/RTCP, SHALL be handled and responded to according to the rules and procedures of [RFC3264] and [RFC3550].
- Media descriptions for a Media Stream of another Media Stream Type SHALL be handled and responded to according to the respective standard for that Media Stream Type.

The SDP handling for disposition notifications SHALL follow the rules and procedures of [RFC5438]. The Media Stream connection SHALL be handled as described in section 5.2.1.4 “*Handling of Media connection parameters for MSRP*”.

An terminating entity that supports the CPM Event Reporting and that has received a SIP request containing the support for CPM Event Reporting as defined in section 6.7 “*CPM Event Reporting Framework*”:

1) SHALL populate in the SDP *a=accept-wrapped-types* attribute of the SIP response the value of the CPM Event Reporting Content-Type as defined in section 6.7.4 “*CPM Event Reporting Data Format*”.

The terminating entity that supports MSRP chunk size negotiation SHALL check the SDP a-line attribute “a=max-chunk-size” to determine if the proposed MSRP chunk size is appropriate for both sending and receiving MSRP messages as per its own configured values based on service provider policies e.g. for the network access type, or the applicable service level agreement.

If an MSRP endpoint is not configured to handle a higher MSRP chunk size, such as a specific value provided by the SDP offerer in the SIP request, or by the SDP answerer in the SIP response, or it does not match its own configured values for its current network access type, then:

– it SHALL use the closest configured MSRP chunk size that is lower than the other endpoint’s value.

5.2.1.4 Handling of Media connection parameters for MSRP

CPM v2.0 entities SHALL handle the Media Stream connection model for MSRP according to the rules and procedures of [RFC4975] and [RFC6714].

The CPM v1.0 and SIMPLE IM v2.0 specifications are based on the MSRP session matching mechanism defined in an expired IETF draft (draft-ietf-simple-msrp-sessmatch), as described in section 5.2.1.4.1 “*Legacy MSRP session matching*” below.

When configured for interoperability of CPM v2.0 with either CPM v1.0 or with SIMPLE IM v2.0 Media Streams based on MSRP, the network entity at the borderline between the two MSRP connection mechanisms MAY decide to act as MSRP B2BUA and apply procedures indicated in this section. This network entity MAY be one of:

- network elements of the SIP/IP Core, or
- the CPM Participating Function (handling different versions of CPM clients, or interoperating with SIMPLE IM clients), or
- the CPM Interworking Function (handling the interworking with a SIMPLE IM network via the SIP/IP Core Network-to-Network Interface).

The network element that is not configured for MSRP interoperability SHALL follow [RFC6714] for the fall-back procedures.

The network element which applies the MSRP interoperability with the MSRP sessmatch mechanism described in section 5.2.1.4.1 "*Legacy MSRP session matching*" on a session leg, SHALL proceed as follows:

1. when acting in a SDP offerer role and if it has knowledge of the MSRP connection model on the forward leg (e.g. based on service policies for interconnect to the destination domain), or when acting in a SDP answerer role and the received SDP for a CPM leg did not contain the SDP 'msrp-cema' attribute,
   a) it SHALL insert its own address information in the SDP "a=path" attribute it sends out, with a valid session-id component in the MSRP URI component of the attribute;
   b) it SHALL set the SDP "setup " attribute, in accordance with [RFC6135];
   c) it SHALL apply session matching procedures on the legacy leg according to expired IETF draft (draft-ietf-simple-msrp-sessmatch), as described in section 5.2.1.4.1 "*Legacy MSRP session matching*" in order to establish the MSRP connection.
2. when acting in a SDP offerer role and has received a SDP answer on a CPM leg that did not contain the SDP 'msrp-cema' attribute, it SHALL apply session matching procedures on the legacy leg according to expired IETF draft (draft-ietf-simple-msrp-sessmatch), as described in section 5.2.1.4.1 "*Legacy MSRP session matching*" in order to establish the MSRP connection.

Other MSRP interworking methods might exist. Such methods are out of scope of this specification. Details of the interworking in the network elements of the SIP/IP Core are also out of scope of this specification. Additional details on the relationship between the IETF RFCs used for MSRP session negotiation and connection establishment can be found in [3GPP TR29.828].

A terminating entity (e.g. a CPM Client, or an IWF, or a CPM Participating Function in a deferral mode) SHALL populate the SDP a=setup attribute in SDP answers as follows:

– CPM Client SHALL use the value of "active";

– Any other terminating entity SHOULD use the value of "passive" according to [RFC4975] and [RFC6135].

5.2.1.4.1 Legacy MSRP session matching

The difference between the session matching mechanism in [RFC4975] and the one defined here is that while the mechanism in [RFC4975] uses the MSRP URI comparison rules for session matching, the expired IETF draft used only the session-id part of the MSRP URI is used.

When an MSRP entity receives the first MSRP request for an MSRP session, the To-Path header field of the request should contain a URI with a session-id part that was provided in the SDP associated with the MSRP session. The entity that accepted the connection looks up the session-id part of the MSRP URI in the received requests, in order to determine which session it matches. The session-id part is compared as case sensitive. If a match exists, the entity shall assume that the host that formed the connection is the host to which this URI was given.

If no match exists, the entity shall reject the request with a 481 MSRP error response. The entity shall also check to make sure the session is not already in use on another connection. If the session is already in use, it shall reject the request with a 506 MSRP error response.

5.3 CPM Conversation Identification

In order to support the concept of CPM Conversations and a threaded view of a CPM Conversation History, three new SIP headers fields have been defined:

- Conversation-ID: this is a header field that identifies the CPM Conversation Identity that is associated with CPM Standalone Messages, CPM File Transfers and CPM Sessions. All CPM Standalone Messages, CPM File Transfers and CPM Sessions belonging to the same CPM Conversation carry the same value for the Conversation-ID header field. All CPM notifications, such as delivery and read reports, that are sent for CPM Chat Messages after the CPM Session was terminated, SHALL carry the CPM Conversation Identity of the respective CPM Session.
- Contribution-ID: this is a header field that identifies the CPM Contribution Identity of an individual CPM Standalone Message, CPM File Transfer or CPM Session and disposition notification that is part of a CPM Conversation.
- InReplyTo-Contribution-ID: this is a header that, in case of a reply to an earlier received CPM Standalone Message, CPM File Transfer or CPM Session, identifies the Contribution-ID of the CPM Standalone Message, CPM File Transfer or CPM Session that is being replied to. A CPM File Transfer sent within a CPM Session uses the value of the Contribution-ID of the CPM Session in the InReplyTo-Contribution-ID header field.

Appendix C "*CPM-defined SIP Headers*" contains a formal definition of these three SIP header fields.

These three SIP header fields are preserved unchanged end-to-end. They SHALL NOT be modified or deleted by any intermediate node in any CPM communication.

The three header fields SHALL be carried in Pager Mode CPM Standalone Messages and disposition notifications (realized over SIP MESSAGE request) and the session invitations of Large Message Mode CPM Standalone Messages, CPM File Transfers and CPM Sessions (SIP INVITE request). All CPM functional components SHALL support these SIP header fields.

5.4 Disposition Notifications

When a CPM User requests to obtain the disposition-state of the sent CPM Standalone Message or of the sent CPM Chat Message or of the sent CPM File Transfer file, the CPM Client SHALL include the appropriate disposition notification request(s) in a CPIM header field of the sent CPM Standalone Message or of the sent CPM Chat Message (in MSRP SEND) or of the sent CPM File Transfer request (i.e. SIP INVITE body).

Disposition notifications defined for CPM are the following:

a) delivery notification, as defined in [RFC5438] and [RFC5438Errata] with the additional clarifications specified in this document; and

b) read report, as defined in [RFC5438] and [RFC5438Errata] with the additional clarifications specified in this document; and

c) interworking notifications extension as specified in Appendix O.

When a delivery notification and/or read report, and/or interworking notification, for one or more CPM Messages or CPM File Transfer file(s) was requested, the receiving entity (e.g. terminating CPM Client, terminating CPM Participating

Function or CPM IWF) SHALL generate the respective disposition notification as described in sect. 5.4.1 "*Generate Delivery Notification*" and 2 "*Generate read Report*" and 5.4.6 "*Generate Interworking Notification*" in any one of the following cases:

1. When a delivery notification was requested and:
   a) when terminating CPM Client:
      i. has successfully received the CPM Message or CPM File Transfer file for which the delivery notification was requested;
   b) Or, when the CPM IWF:
      i. has received a successful delivery notification for interworked CPM Message(s) or CPM File Transfer file(s).
2. When a display notification was requested and:
   a) When terminating CPM Client:
      i. has successfully rendered the CPM Chat Message or the CPM File Transfer file for which the disposition notification was requested, whether it was delivered within the CPM Session, or later on, as a stored CPM Message or a stored CPM File Transfer file.
3. When an interworking notification was requested and:
   a) when the CPM IWF:
      i. has received a successful delivery notification for interworked CPM Message(s) or CPM File Transfer file(s).

A disposition notification SHALL NOT be requested for a sent disposition notification as specified in [RFC5438].

The disposition notification mechanism in CPM is an extension to the specification in [RFC5438] and [RFC5438Errata]. The extension is that an original message requesting disposition notification can be also sent using a MSRP SEND request in case of Large Message Mode CPM Standalone Message and of CPM Chat Message, and of CPM File Transfer when disposition notification is sent within the CPM Session.

The disposition notification message is sent using either:

i. SIP MESSAGE method according to the rules and procedures of [RFC5438] and [RFC5438Errata] , for CPM Standalone Messages and CPM File Transfer(s); or for CPM Sessions when the disposition notifications are sent after the CPM Session was terminated; or,

ii. MSRP procedure, when the disposition notifications are sent during an on-going CPM Session with the requestor, reusing the same MSRP session.

When SIP MESSAGE method is used, or when a session is set up explicitly to deliver stored notifications, the disposition notifications are sent back to the user that has originated the message for which the disposition notifications are being issued.

The CPM Client that receives a disposition notification SHALL process the received IMDN payload, as follows:

1) SHALL check if a <delivery-notification> element is present for each given <message-id> in the IMDN notification;
   a) If the element is present, then it SHALL check the child element of the <status> sub-element;
      i. If the <delivered/> element is found, then the CPM Client SHALL determine that the respective CPM Message, CPM Chat Message or CPM File Transfer identified by the <message-id> was successfully delivered to the recipient CPM User's Client(s) and SHALL continue with procedures in section 5.4.3 "*Receive Delivery Notification*";
      ii. If any other element is found, then the CPM Client SHALL return a 200 OK response and SHALL determine that the delivery has failed.
2) SHALL check if a <display-notification> element is present for each given <message-id> in the IMDN notification;
   a) If the element is present, then it SHALL check the child element of the <status> sub-element;
      i. If the <displayed/> element is found, then the CPM Client SHALL determine that the respective CPM Message, CPM Chat Message or CPM File Transfer identified by the <message-id> was successfully read by the recipient CPM User and SHALL continue with procedures in section 5.4.4 "*Receive Display Notification*";

ii. Otherwise, if other elements are found, then the CPM Client SHALL return a 200 OK response and ignore this IMDN notification;

3) SHALL check if an < interworking-notification> element is present for each given <message-id> in the IMDN notification;

   b) If the element is present, then it SHALL check the child element of the <status> sub-element;

      i. If the <legacy-sms/> element as defined in Appendix O is present:

         1. then the CPM Client SHALL determine that the CPM Message, CPM Chat Message or CPM File Transfer identified by the <message-id> was successfully delivered to the recipient CPM User via non-CPM technology i.e. SMS;

         2. to support setting the appropriate service expectations, the CPM Client SHOULD inform the CPM User that not all CPM features offered for the CPM Message, or CPM Chat Message or CPM File Transfer are available (e.g. if a display notification was also requested, this SHALL not be received since it's not supported via SMS, or in case of a CPM Chat Message that an 'isComposing' notification SHALL not be expected either);

         3. SHALL continue with procedures in section 5.4.7 "*Receive Interworking Notification*".

      ii. If the <legacy-mms/> element as defined in Appendix O is present:

         1. then the CPM Client SHALL determine that the CPM Message, CPM Chat Message or CPM File Transfer identified by the <message-id> was successfully delivered to the recipient CPM User via non-CPM technology i.e. MMS;

         2. to support setting the appropriate service expectations, the CPM Client SHOULD inform the CPM User that not all CPM features offered for the CPM Message, or CPM Chat Message or CPM File Transfer are available (e.g. in case of a CPM Chat Message, that an 'isComposing' notification SHALL not be expected);

         3. and SHALL continue with procedures in section 5.4.7 "*Receive Interworking Notification*".

      iii. Otherwise, if other elements are found, then the CPM Client SHALL return a 200 OK response and ignore this IMDN notification;

4) Otherwise, if none of the <delivery-notification> or <display-notification> or <interworking-notification> elements are present, then it SHALL return a 200 OK response and ignore this IMDN notification.

5.4.1 Generate Delivery Notification

Upon receiving a CPM Standalone Message, or a CPM Chat Message, or CPM File Transfer file request, the receiving entity which assumed the responsibility to generate the delivery notification (e.g. terminating CPM Client, terminating CPM Participating Function or CPM IWF), SHALL perform the following:

1. SHALL check whether the message or the CPM File Transfer request (i.e. SIP INVITE body) contains the request for a delivery notification element in a CPIM header as described in [RFC5438];
2. If true, the receiving entity SHALL construct a delivery notification (IMDN) according to the rules and procedures of [RFC5438] and [RFC5438Errata], with the following additional clarifications:
   a. If any IMDN-Record-Route header fields were received in the corresponding CPM Standalone Message, or in the corresponding CPM Chat Message, the receiving entity SHALL include each in IMDN-Route header fields preserving same order (i.e. IMDN-Record-Route header fields SHALL NOT be present in the generated IMDN message), as specified in [RFC5438];
   b. the CPIM message-body [RFC3862] carries an IMDN XML document formatted, as specified in [RFC5438];
   c. the Content-Type header field SHALL be set to "message/imdn+xml" as specified in [RFC5438];
   d. the <message-id> element SHALL be set to the IMDN Message-ID value of the CPIM header of the original CPM Chat Message or CPM File Transfer request for which this IMDN notification is issued;
   e. the <delivery-notification> element SHALL be populated as per the disposition type that the sender requested and is being reported;

   f. Except for the case of 1-1 CPM Session, the CPIM "From" header field SHALL contain the authenticated address of the CPM User as defined in section 6.1 "Authenticated Originator's CPM Address", on behalf of which the receiving entity is issuing the disposition notification;
   g. Except for the case of 1-1 CPM Session, the CPIM "To" header field SHALL contain the authenticated CPM address of the disposition notification requestor, as described in step 2.a.i.b above;
   h. In the case of the 1-1 CPM Session, both the CPIM "From" and CPIM "To" header fields SHOULD be set to "sip:anonymous@anonymous.invalid" to prevent revealing the CPM User's identity when transmitted over unprotected links. A CPM Client receiving a CPIM message in a 1-1 CPM Session SHOULD therefore ignore the identity indicated in the CPIM headers;
   i. A disposition notification SHALL NOT be requested for a sent disposition notification as specified in [RFC5438].

3. The receiving entity SHALL send this notification:
   a. Using SIP MESSAGE method for:
      i. a CPM Standalone Message,
      ii. a CPM File Transfer,
      iii. if no active session exists with the CPM User who sent the message received in a CPM 1-1 Session, and
      iv. if no active session exists for the CPM Group Session in which the message was received, for the CPM Group Session case;

   In this case, the receiving entity:
      i. SHALL populate the Request-URI with the authenticated originator's CPM Address of the received SIP MESSAGE or SIP INVITE request, as specified in section 6.1 "*Authenticated Originator's CPM Address*";

      ii. SHALL include, for IMDNs associated with CPM Messages exchanged in a Group communication (CPM Group Standalone Message, CPM Group Chat Message or CPM Group File Transfer request), in the Conversation-ID header field the value received in the original CPM Standalone Message, CPM File Transfer or of the CPM Session, and a Contribution-ID header field which MAY have the same value as the Contribution-ID value of the original SIP request, or which MAY contain a newly generated value;
      iii. SHOULD include, for IMDNs associated with CPM Messages exchanged in a 1-1 communication (1-1 CPM Standalone Message, 1-1 CPM Chat Message or 1-1 CPM File Transfer request), a Conversation-ID header field and a Contribution-ID header field with newly generated values;
      iv. SHALL populate an Accept-Contact header field set to the original CPM Feature Tag value (defined in Appendix H) of the CPM request (i.e. the ICSI of the corresponding service percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 "*Coding of the ICSI*") that included the CPM Message or CPM File Transfer for which the IMDN is being generated;
      v. If the receiving entity generating the delivery notification is:
         a. the CPM Client, it SHALL populate an P-Preferred-Service header field set to the P-Asserted-Service header value of the CPM request that included the CPM Message or CPM File Transfer for which the IMDN is being generated;
         b. the CPM Participating Function or CPM Interworking Function, it SHALL populate an P-Asserted-Service header field set to the original P-Preferred-Service or the P-Asserted-Service header value of the CPM request that included the CPM Message or the CPM File Transfer for which the IMDN is being generated;
      vi. the "Content-type" header field SHALL contain a MIME type "message/cpim" as specified in [RFC3862];
      vii. SHALL include as SIP Message payload the IMDN generated in step 2:
      viii. SHALL send the SIP MESSAGE request carrying a delivery notification according to the rules and procedures of the SIP/IP core.

b. Using MSRP procedure in an active CPM Session if such an active session exists with the sender of a message received in a CPM 1-1 Session or for the CPM Group Session for the Group to which the CPM Message relates:

1. SHALL construct a MSRP SEND toward the requestor of the disposition notification with the following clarifications:
   i. the “Content-type” header field SHALL contain a MIME type “message/cpim” as specified in [RFC3862];
   ii. the “From-Path” and “To-Path” header fields SHALL be set as described in [RFC4975] and [RFC6714].
   iii. SHALL include as MSRP payload the IMDN generated in step 2:

5.4.2 Generate Read Report

Upon receiving a CPM Standalone Message or a CPM Chat Message or a CPM File Transfer and if privacy settings allow it, the CPM Client:

1. SHALL check whether the message contains the request for display notification element in CPIM header, as described in [RFC5438], to generate a read report;
   a. If it contains the request for display notification:
      i. If the message or file is already stored in the CPM User’s network-based Message Storage, the CPM Client SHALL check if the \Seen flag of the corresponding stored object is set.
         - If it is not set, proceed with step 2;
         - Otherwise, no further steps are required.
      ii. Otherwise, proceed with step 2.
   b. Otherwise, if no display notification is requested no further steps are required.
2. When the CPM Client determines that the CPM User has read the message or file (e.g. by user confirmation) and allowed by local device’s settings, the CPM Client SHALL construct a read report according to the rules and procedures of [RFC5438] and [RFC5438Errata], with the following additional clarifications:
   a. If any IMDN-Record-Route header fields were received in the corresponding CPM Standalone Message, or in the corresponding CPM Chat Message, the receiving entity SHALL include each in IMDN-Route header fields preserving same order (i.e. IMDN-Record-Route header fields SHALL NOT be present in the generated IMDN message), as specified in [RFC5438];
   b. the CPIM body [RFC3862] carries a IMDN XML document formatted, as specified in [RFC5438];
   c. the Content-Type header field SHALL be set to =” message/imdn+xml “as specified in [RFC5438];
   d. the <message-id> element SHALL be set to the IMDN Message-ID value of the CPIM header of the original CPM Chat Message or the CPM File Transfer file for which this IMDN notification is issued;
   e. the <display-notification> element SHALL be set as per the disposition type that the sender requested and that is being reported.
3. The CPM Client SHALL send this read report:
   a. Using SIP MESSAGE method for:
      i. a CPM Standalone Message,
      ii. a CPM File Transfer,
      iii. if no active session exists with the user who sent the message for a message received in a CPM 1-1 Session and
      iv. if no active session exists for the Group Chat in which the message for a message received in a CPM Group Session

   In this case:

   1) SHALL populate the SIP and CPIM header fields in the same way as described for delivery notification, specified in section 5.4.1. “*Generate Delivery Notification*”, with the difference that for Read Reports only the <display-notification> element SHALL be populated as described in step 2., as per the disposition type that the sender requested and is being reported;

2) SHALL send the SIP MESSAGE request carrying a read report according to the rules and procedures of the SIP/IP core.
3) if a Message-UID header field is present in the 200 “OK” SIP response indicating that the CPM Participating Function has stored the SIP MESSAGE containing the display report IMDN, the CPM Client SHALL retrieve the stored IMDN message object UID from the Message-UID header field and keep it locally.

NOTE: The storage of the IMDN display report message object in the CPM Message Store indicates that future displays of the associated user message to the CPM User need not generate a read report (from any of the CPM User Clients).

f. Using MSRP procedure during an active CPM Session if an active session exists with the sender of a message received in a CPM 1-1 Session or for the CPM Group Session for the Group to which the CPM Message relates:

1) It SHALL construct a MSRP SEND toward the requestor of the disposition notification as described for delivery notification, specified in section 5.4.1. “*Generate Delivery Notification*” via MSRP method, with the difference that for Read Reports only the <display-notification> element SHALL be populated as per the disposition type that the sender requested and is being reported.
   i. The “From” and “To” header field SHALL be set according to [RFC4975] following the parameters negotiated during the setup of the session.

5.4.3 Receive Delivery Notification

Upon receiving a delivery notification, the CPM Client SHALL process the notification as specified in [RFC5438] and [RFC5438Errata].

In a multi-device scenario, if a sender receives more than one delivery notification for a sent message, it SHALL discard all copies except the first one it receives.

The CPM Participating Function SHALL process received delivery notifications according to the procedures described in section 8.2.4 “*Receiving a Disposition Notification*”.

5.4.4 Receive Read Report

Upon receiving a read report, the CPM Client SHALL process the notification as specified in [RFC5438] and [RFC5438Errata].

In a multi-device scenario, if a sender receives more than one read report (displayed notification) for a sent message, it SHALL discard all copies except the first one it receives.

The CPM Participating Function SHALL process received read report notifications as described in section 8.2.4 “*Receiving a Disposition Notification*”.

5.4.5 Multidevice handling

In multi-device scenarios, the CPM IMDN Notification sent via SIP MESSAGE is forked to the suitable CPM Client(s) of the CPM User that has requested it, except for the read reports which are not delivered to a suitable CPM Client with only SMS capable connectivity.

When a CPM Standalone Message, or a CPM Chat Message, or a CPM File Transfer request, containing a disposition notification request is targeted at multiple recipients (i.e. pre-defined/ad-hoc group) or multiple different types of disposition notifications are requested for the same CPM Standalone Message or the same CPM Chat Message or the CPM File Transfer file, the originating user MAY receive aggregated disposition notifications.

If none of the originator CPM User’s devices is available at the time of delivery of the IMDN disposition notification, the CPM Participating Function SHALL defer the disposition notification for later delivery to the CPM User. In such a case, the CPM Participating Function SHALL follow one of the following procedures:

- CPM Standalone Message deferral and delivery, as described in sect. 8.3.1.6 “*Defer CPM Standalone Messages*” for deferred delivery of IMDNs via SIP MESSAGE method to the registered suitable CPM Clients of the CPM User; or

- 1-1 CPM Session deferred delivery, as described in section 8.3.2.9.1 “*Deferred delivery initiated by the CPM Participating Function*” for deferred delivery of IMDNs via MSRP together with the deferred chat messages belonging to that CPM Session.

In case of delivery to multi-device on the recipient side, it MAY be possible for the originating CPM Client to receive a delivery notification, an interworking notification and/or a read notification for that same CPM Message or CPM File Transfer. The CPM Client SHALL decide how to inform the CPM User in this case, based on service provider policies.

5.4.6 Generate Interworking Notification

Upon receiving a CPM Standalone Message, or a CPM Chat Message, or CPM File Transfer file request, the receiving entity which assumed the responsibility to generate the interworking notification (i.e., the terminating CPM IWF):

1. SHALL check whether the message or the CPM File Transfer request (i.e. SIP INVITE body) contains the request for an interworking notification element in a CPIM header;
2. If true the receiving entity SHALL construct the notification (IMDN) according to the rules and procedures of [RFC5438] and [RFC5438Errata], with the following additional clarifications:
    a. If any IMDN-Record-Route header fields were received in the corresponding CPM Standalone Message, or in the corresponding CPM Chat Message, the receiving entity SHALL include each in IMDN-Route header fields preserving same order (i.e. IMDN-Record-Route header fields SHALL NOT be present in the generated IMDN message), as specified in [RFC5438];
    b. the CPIM message-body [RFC3862] carries an IMDN XML document formatted, as specified in [RFC5438];
    c. the Content-Type header field SHALL be set to “message/imdn+xml” as specified in [RFC5438];
    d. the <message-id> element SHALL be set to the IMDN Message-ID value of the CPIM header of the original CPM Chat Message or CPM File Transfer request for which this IMDN notification is issued;
    e. the <interworking-notification> element SHALL be populated as requested by the sender and based on the interworking performed by CPM IWF. The CPM IWF SHALL further populate its sub-elements depending on the technology used for the interworked delivery:
        i. if the CPM Message was successfully delivered via interworking to SMS, then it SHALL populate the <legacy-sms> element, as described in section 6.2.4. “Successful SMS Delivery” of [OMA-CPM_TS_IWF];
        ii. if the CPM Message or the CPM File Transfer request was successfully delivered via interworking to MMS, then it SHALL populate the <legacy-mms> element;
    f. The CPIM “From” header field SHALL contain the authenticated address of the CPM User as defined in section 6.1 “Authenticated Originator's CPM Address”, on behalf of which the receiving entity is issuing the interworking disposition notification;
    g. SHALL set the CPIM To header field, using the authenticated originator’s CPM Address of the received SIP MESSAGE or SIP INVITE request, as defined in section 6.1 “Authenticated Originator's CPM Address”.
3. The receiving entity SHALL send this notification:
    a. Using SIP MESSAGE method for:
        i. a CPM Standalone Message,
        ii. a CPM File Transfer,
        iii. if no active session exists with the user who sent the message for a message received in a CPM 1-1 Session and
        iv. if no active session exists for the Group Chat in which the message for a message received in a CPM Group Session.

        In this case, the receiving entity:

        v. SHALL populate the Request-URI with the authenticated originator’s CPM Address of the received SIP MESSAGE or SIP INVITE request, as specified in section 6.1 “*Authenticated Originator's CPM Address*”:
        vi.

i. SHALL include, for IMDNs associated with CPM Messages exchanged in a Group communication (CPM Group Standalone Message, CPM Group Chat Message or CPM Group File Transfer request), in the Conversation-ID header field the value received in the original CPM Standalone Message, CPM File Transfer or of the CPM Session, and a Contribution-ID header field which MAY have the same value as the Contribution-ID value of the original SIP request, or which MAY contain a newly generated value;

vii. SHOULD include, for IMDNs associated with CPM Messages exchanged in a 1-1 communication (1-1 CPM Standalone Message, 1-1 CPM Chat Message or 1-1 CPM File Transfer request), a Conversation-ID header field and a Contribution-ID header field with newly generated values;

viii. SHALL populate an Accept-Contact header field set to the original CPM Feature Tag value of the CPM request that included the CPM Message or CPM File Transfer for which the IMDN is being generated;

ix. SHALL populate an P-Asserted-Service header field set to the original P-Preferred-Service or the P-Asserted-Service header value that was included in the CPM Message or the CPM File Transfer, for which the IMDN is being generated;

x. SHALL send the SIP MESSAGE request carrying the interworking notification according to the rules and procedures of the SIP/IP core and [3GPP TS24.229].

b. Using MSRP procedure during an active CPM Session if an active session exists with the sender of a message received in a CPM 1-1 Session or for the CPM Group Session for the Group to which the CPM Message relates.

In this case the receiving entity SHALL construct a MSRP SEND request toward the requestor of the disposition notification with the following clarifications:

i. the “Content-type” header field SHALL contain a MIME type “message/cpim” as specified in [RFC3862];

ii. the “From-Path” and “To-Path” header fields SHALL be set as described in [RFC4975] and [RFC6714].

iii. SHALL include the interworking notification constructed in step 2 as MSRP payload.

1.

5.4.7 Receive Interworking Notification

Upon receiving an interworking notification, the CPM Client SHALL process the notification as specified in [RFC5438] and [RFC5438Errata], with the extensions defined in Appendix N of this document.

Whenever a CPM Client receives an interworking notification then the CPM Client SHALL interpret that the CPM Message or CPM File Transfer has only been delivered via interworking to the recipient and SHOULD inform the CPM User that he/she cannot benefit from the full CPM end to end features (e.g. some CPM features are not available over legacy messaging, such as “isComposing” for CPM Chat Messages should not be expected from the other party).

The CPM Participating Function SHALL process received interworking notifications according to the procedures described in section 8.2.4 “*Receiving a Disposition Notification*”.

5.5 “isComposing” Notifications

The "i*sComposing*" notification is generated and processed according to the rules and procedures of [RFC3994]. Consequently, the ‘*isComposing*’ indication for CPM Chat Message(s) exchanged during 1-1 CPM Session is not sent with CPIM headers, and a delivery and/or displayed notification can thus not be requested.

The ‘*isComposing*’ indication sent for CPM Chat Messages exchanged during a CPM Group Session SHALL contain CPIM header fields as per recommendation of [RFC3994], to indicate the originator of the ‘*isComposing*’ message in the “From” header field and a delivery and/or displayed notification SHALL NOT be requested.

The ‘*isComposing*’ indications are never stored in the CPM Message Store.

The CPM Client SHALL stop sending “isComposing” notifications to another CPM User in a CPM Session, after it received from that CPM User an interworking IMDN notification for a CPM Chat Message in that CPM Session. The client SHALL resume sending isComposing notifications once it receives at least one delivery or display notification.

5.6 CPM Service IDs

The CPM service capabilities may be exchanged through different methods.

For CPM service capabilities publication through OMA Presence Enabler [OMA PRES], the following <service-description> child elements, registered by OMNA, SHALL be used in the presence document, according to [OMA PDE]:

For CPM Standalone capabilities:

→<service-id>→org.openmobilealliance:StandaloneMsg

→<version>→2

For CPM Session capabilities:

- CPM Session:

  →<service-id>→org.openmobilealliance:ChatSession

  →<version>→2

For CPM File Transfer capabilities:

→<service-id>→org.openmobilealliance:File-Transfer

→<version>→2

→<service-id>→org.openmobilealliance:File-Transfer-thumb

→<version>→2

6. Common Procedures

6.1 Authenticated Originator’s CPM Address

The authenticated originator's CPM Address is:

- the CPM Address of the originating CPM Client that has been authenticated by the SIP/IP core i.e. one of the addresses received in the P-Associated-URI header field of the last successful registration response received from the SIP/IP Core; or
- the CPM Pre-defined Group address when the CPM Controlling Function sends a CPM Standalone Message, CPM Session Invitation or CPM File Transfer request to CPM Pre-defined Group members; or
- the CPM Group Session Identity for this particular CPM Ad-hoc Group when the CPM Controlling Function invites users to a CPM Ad-hoc Group.

When the SIP/IP core corresponds to 3GPP/3GPP2 IMS, then the authenticated originator's CPM Address is contained in the P-Asserted-Identity header field according to the rules and procedures of [3GPP TS 24.229] / [3GPP2 X.S0013.004] and as described in [RFC3325] and [RFC5876]. The CPM Client MAY insert a P-Preferred-Identity header field, which contains the CPM Client’s preferred identity, for the SIP/IP core to be used inside the P-Asserted-Identity header field. The P-Preferred-Identity is selected by the CPM Client from the list of CPM Addresses in the P-Associated-URI header field received in the last successful registration response.

If anonymity is required, the From header field SHALL contain an anonymous URI according to [RFC3323] and optionally a Pseudonym or "Anonymous" as the display name, and the Privacy header field values SHALL be set according to [RFC3323] and [RFC3325].

NOTE: The term “anonymity” in this specification is referring to the term “privacy” used in [RFC3323].

If the CPM Participating Function cannot obtain an authenticated originator's CPM Address for an initial request, it SHALL reject the request with a SIP 403 "Forbidden" response. The CPM Participating Function SHOULD include a SIP Warning header field to explain the reason in a human readable textual form.

When the Referred-By header field is set to the authenticated originator’s CPM Address and SHALL follow the syntax described in [RFC3892], with following additions and clarifications:

- the value populated in the Referred-By header field SHALL be set according to service provider's policy, which includes one of the following options:
    a) the SIP URI value from the P-Asserted-Identity header field [RFC3325]; or
    b) the TEL URI value from the P-Asserted-Identity header field [RFC3325]; or
    c) either one of the SIP URI or TEL URI values from the P-Asserted-Identity header field [RFC3325], with a new parameter named “add-refs” including the remaining identity found in the P-Asserted-Identity header field;
- the optional ‘cid’ parameter defined in [RFC3892] is not used in this specification.

6.1.1 Identifying the sending user and device in SIP requests and responses

If a CPM Client obtains GRUUs from the registrar, as per [RFC5627], and the CPM request needs a device identifier, then the CPM Client SHALL use the public GRUU as the URI to be used in the Contact header field in non-REGISTER SIP requests and responses that it emits. For example, the GRUU URI SHALL be included in the Contact header field of any SIP INVITE request and related SIP 200 OK response.

Since it is not mandatory for the SIP/IP Core to support GRUU, then if a CPM Client does not obtain a GRUU from the registrar, but it needs to use a device identifier in the CPM request, then it SHALL include the +sip.instance feature tag in the Contact header field with the same instance identifier value used at registration in any non-REGISTER SIP request and responses that it emits, and which carry a Contact header field.

For SIP MESSAGE requests, the identity of the sender does not carry a device identifier and is populated in the P-Asserted-Identity header field and which SHALL include:

- the authenticated originator's CPM Address of the sender.

In MSRP messages (e.g., in CPM Chat Messages sent within a CPM Session), the sender sets the CPIM From header field in the MSRP SEND request to the authenticated CPM Address for the sender CPM User, as received from the SIP/IP Core in the P-Associated-URI header field received in the last successful registration response.

6.1.2 Identifying the recipient device in SIP requests and responses

a) A CPM Participating Function MAY have to target a SIP request to a specific client. If the SIP/IP core supports GRUU and the CPM Participating Function is aware of the targeted CPM Client’s public GRUU, the CPM Participating Function SHALL set:the Request-URI of the generated SIP request to the Authenticated Originator's CPM Address as specified in section 6.1 “*Authenticated Originator's CPM Address*” (i.e. based on either the P-Asserted-Identity, or on the Referred-By header fields), of the CPM User from the incoming SIP request, together with the GRUU related “gr=” URI parameter. If no SIP URI is available then the TEL URI SHALL be used.

If the SIP/IP core does not supports GRUU or the CPM Participating Function is not aware of the device’s public GRUU then if it needs to ensure that a generated SIP request is sent back to a particular device, a new Accept-Contact header field is added by the CPM Participating Function, carrying only the +sip.instance parameter and instance identifier value as well as the tags “explicit” and ”require” described in [RFC3841].

The CPM Client supporting the *+sip.instance* procedures SHALL respond to the SIP request with a 486 BUSY HERE if the identifier value of the *sip.instance* parameter included in the *Accept-Contact* header of that incoming SIP request does not match theirs.

In context of multi-client and multi-device deployments, it is possible that same CPM User identity is shared by the CPM Client and by other IMS Clients (e.g. Voice over LTE client using SMS over IP). To ensure that CPM requests are forked only to CPM Clients (i.e. which have explicitly registered with the required CPM capabilities), a dedicated Accept-Contact header field SHALL be added to all CPM SIP requests carrying the CPM ICSI defined for the respective CPM service with the “*require”* and “*explicit”* parameters described in [RFC3841], by the terminating CPM Participating Function.

6.2 SIP/IP Core

The CPM Functional components SHALL interface to the underlying SIP/IP core in accordance with the rules and procedures for that SIP/IP core as described in [RFC3261].

When the SIP/IP core corresponds with 3GPP/3GPP2 IMS, these rules and procedures are described in [3GPP TS 24.229] / [3GPP2 X.S0013.004].

6.3 Display Name and Anonymity

Using a display name in a CPM Conversation serves different purposes. In case anonymity is requested, it provides the user with a Pseudonym, hiding the user’s real identity. In case anonymity is not requested, it is used to display a human readable name for the user. Finally, for a CPM Conversation in a CPM Pre-defined Group, each Participant may be given a group specific display name. This group specific display name is stored in the Group XDMS, as specified in [OMA-XDM-Group].

The CPM Participating Function, CPM Controlling Function and CPM Client MAY send a display name in SIP requests and SIP responses.

The CPM Client MAY support the use of a display name. The CPM Client MAY support the user agent behaviour described in [RFC3323] and [RFC3325].

The CPM Participating Function and CPM Controlling Function SHALL support the use of a display name and the network-provided anonymity/privacy described in [RFC3323] and [RFC3325].

The following text describes the procedures when the CPM Client, the CPM Participating Function and CPM Controlling Function support the use of a display name and anonymity.

A CPM Client MAY provide the inviting CPM User's Pseudonym (this is a user defined name) in the "display-name" part of the authenticated originator's CPM Address, i.e. in the P-Preferred-Identity header and in the From header of the initial SIP INVITE request or SIP REFER request sent towards the CPM Participating Function.

NOTE: A display name included in the P-Preferred-Identity header is moved to the P-Asserted-Identity header by the underlying SIP/IP core if validation of the CPM Address is successful as described in [RFC3325].

The terminating CPM Client MAY provide the invited CPM User's display name in the "display-name" part of the To and, if included, P-Preferred-Identity header fields of a SIP 200 “OK” response to an initial SIP INVITE request.

An originating or terminating CPM Client MAY request anonymity by adding privacy type “id” to the Privacy header as described in [RFC3323] and [RFC3325].

The CPM Participating Function SHALL act on privacy type “id” detected in a Privacy header as described in [RFC3323] and [RFC3325] to provide the CPM User with anonymity when requested.

The CPM Controlling Function SHALL determine the display name to use, according to the following priority order:

1. In case anonymity is requested, and the CPM Pre-defined Group definition as specified in [OMA-XDM-Group] allows anonymity, the display name as provided by the CPM Client SHALL be used if the display name is allowed by service provider's policy. If no display name was defined by the CPM Client, or this display name is already in use in the session, the CPM Controlling Functions chooses a display name as defined in 9.2.13 “*Pseudonyms in a CPM Group Session*”.
2. In case anonymity is not requested, the display name received in the “display-name” part of the authenticated originator's CPM Address of initial SIP requests and, for display name of the responding side, of SIP 200 “OK” responses to the initial SIP requests from Participants SHALL be used. If no authenticated originator's CPM Address was provided or it didn't include a display name, the display name received in the “display-name” part of the From header of initial SIP requests and, for display name of responding side, in the “display-name” part of the To header of the SIP 200 “OK” responses to the initial SIP requests from Participants SHALL be used.

A CPM Client receiving a display name (either on the terminating or the originating side) SHALL display the display name to the user.

6.4 SIP Warning Header Field

6.4.1 General

The CPM Participating Function or CPM Controlling Function MAY include a free text string in an error response to SIP requests.

When the CPM Participating Function or CPM Controlling Function includes a text string in an error response to a SIP INVITE request or a SIP MESSAGE request, a SIP Warning header field as specified in [RFC3261] SHALL be included and the Warning code SHALL be set to 399.

An appropriated warning text string as defined in section 6.4.2 SHALL be included with the SIP Warning header and MAY include the host name set to the host name of the CPM Participating Function or CPM Controlling Function.

The CPM Client MAY include the preferred language in the Accept-Language header field in the SIP INVITE request, SIP MESSAGE request or SIP 2xx response.

The CPM Participating Function or CPM Controlling Function SHOULD choose the language of the warning text in the SIP Warning header field depending on the preferred language indicated in the Accept-Language header field received from the

CPM Client in the SIP INVITE request, SIP MESSAGE request or in the SIP 2xx response. If the warning text is to be translated, only explanatory text of the free text string SHALL be replaced by the preferred language.

6.4.2 Warning Texts

The warning text string included in a SIP Warning header field consists of an explanatory text preceded by a 3-digit text code, according to the format <text code> + <explanatory text>, for example “102 Too many participants”.

The warning texts and text codes are defined in Table 1.

Text Code	**Explanatory text**	**Description**
102	Too many participants	The maximum number of Participants allowed in a CPM Session is exceeded.
105	Isfocus already assigned	A conference focus (a CPM Controlling Function) has already been assigned to the CPM Session.
119	Anonymity not allowed	Anonymity is requested, but is not allowed.
122	Function not allowed	Function is not allowed, but a detailed description about the reason is not given.
123	Session does not exist	The target session in the Request URI does not exist.
125	No messages	Messages are requested to be retrieved, but there are no messages.
127	Service not authorised	The User is not authorised for this service.
129	No destinations	No destination addresses available for the action.
132	Version not supported	The CPM version indicated in the request is not supported.
133	Size exceeded	The file size is bigger than the acceptable file size for a CPM File Transfer.
134	Closed group chat not supported	The CPM Closed Group Session is not supported.

Table 1: CPM specific warning texts and text codes

6.5 Communicating with the ISF and IWF

In order to send a CPM Standalone Message, CPM File Transfer or CPM Session Invitation to the ISF, the CPM Participating Function or the CPM Controlling Function SHALL act as a UAC, as defined in [RFC3261], and send the SIP MESSAGE request (corresponding with a Pager Mode CPM Standalone Message or a CPM Standalone Message Disposition Notification) or SIP INVITE request (corresponding with a Large Message Mode CPM Standalone Message, CPM File Transfer or a CPM Session Invitation) directly to the ISF, without routing the SIP request via the SIP/IP core.

6.5.1 Interworking Response handling

Upon receiving a SIP 200 "OK" response from IWF to a SIP IMESSAGE request, the CPM Participating Function:

1. SHALL check if the following SIP header fields are present:
   A. If a Message-Context SIP header field is present (i.e. when successful delivery via either SMS or MMS was performed), then it SHALL use its value to store it into a Message-Context MIME header for the stored message object in the CPM Message Store;
   B. If a Messge-Correlator SIP header is present (i.e. when successful delivery via MMS was performed), then it SHALL use its value to store it into a Message-Correlator MIME header for the stored message object in the CPM Message Store;
2. SHALL proceed to record the message into the CPM Message Store, as described in section 8.5.4 "*Record CPM Messages delivered via interworking*".

Upon receiving a SIP 200 "OK" response from IWF to a SIP INVITE request, the CPM Participating Function is acting as a B2BUA:

1. SHALL check whether the all offered Media Streams have been accepted;
2. If not, the CPM Participating Function MAY attempt to establish a new session for the rejected Media Streams. To establish the new session the CPM Participating Function:
   a. SHALL check if service provider policy allows to establish a new session for the new Media Stream;
   b. If allowed, SHALL generate a SIP INVITE request according to the rules and procedures of [RFC3261] with the following clarifications;
      i. SHALL include SDP parameters about the rejected Media Streams in the SDP body as described in section 5.2.1.2 "*SDP Handling at Intermediate Nodes*" and information received in the SDP from the originating client;
      ii. SHALL send the SIP INVITE request directly to the ISF as described in section 6.5 "*Communicating with the ISF and IWF"*.
3. Otherwise, SHALL perform the same procedures as receiving a SIP 200 "OK" response from a CPM Client and SHALL start listening for an incoming MSRP session packets before sending the SIP ACK acknowledge the SIP 200 "OK" response;
4. Once Media Plane is established with CPM IWF, if the CPM Participating Function receives:
   a. an MSRP 200 "OK" response, and if the respective CPM Message, CPM Session or CPM File Transfer needs to be recorded, then it SHALL execute the procedures specified in section 8.5.4 "*Record CPM Messages delivered via interworking*";
   b. an MSRP 413 error response for a MSRP SEND request, then it SHALL check the service provider policy about handling undelivered messages, to determine if deferred services have to be triggered, as described in section 8.3.6 "*Delivery Policies in Terminating CPM Participating Function*";
   c. any other MSRP response, it SHALL handle it as specified in [RFC4975].

Upon receiving a SIP 480 "Temporarily Unavailable" response from ISF for a SIP INVITE or a SIP MESSAGE request, the CPM Participating Function SHALL check the service provider policy about handling undelivered messages, as described in section 8.3.6 "*Delivery Policies in Terminating CPM Participating Function*".

6.6 Suitable CPM Clients

A CPM User MAY have multiple devices, hence multiple CPM Clients. One of the CPM User's devices MAY be identified as Primary Device. All other devices (zero or more) are considered Secondary Devices.

The distinction between the CPM Client on the Primary Device and the CPM Client on a Secondary Device is made based on the information provided by each CPM Client in the registration.to the SIP/IP Core.

A CPM Client SHALL be considered suitable when:

- the client capabilities indicate support for the CPM technology needed for the specific request, i.e. Pager Mode CPM Standalone Message, Large Message Mode CPM Standalone Message, CPM Session, CPM File Transfer, or support for deferred delivery or system messages, whichever is applicable, and
- the client connectivity permits use of CPM requests, or the connectivity of a Primary Device permits receiving Interworked CPM requests, and
- if server provider policies exist, then they allow delivering the specific CPM request or Interworked CPM requests, to that CPM Client of the CPM User, and
- if CPM User preferences are used, then they also allow the delivery of the specific CPM request, or Interworked CPM requests, to that CPM Client of the CPM User, and
- if the CPM Participating Function requires a specific User Agent version to be supported and the OMA CPM release version of the CPM Client as specified in Appendix D "*Release Version in User-agent and Server headers*" supports it.

6.7 CPM Event Reporting Framework

CPM Event Reporting Framework provides procedures and interfaces of the CPM Enabler to allow:

- A CPM Client to inform the CPM Participating Function of one or more events that have occurred on that CPM Client or device;
- A CPM Client to pass information about event(s) occurred on the device to an Application Server in the network, via the CPM Participating Function;
- The CPM Participating Function to further inform the other CPM Clients on CPM User devices, of the event(s) reported to have occurred on one of the CPM Clients of the CPM User;
- The CPM Participating Function, or an Application Server in the network via the CPM Participating Function, to provide to the CPM Client appropriate information in response to the event reported by the CPM Client.
- The CPM Participating Function, or an Application Server in the network via the CPM Participating Function, to request information about event(s) occurred on a CPM Client or device.
- A CPM Client to provide to the CPM Participating Function, or an Application Server in the network via the CPM Participating Function, the information about requested event(s).

6.7.1 Service Identification

The CPM Event Reporting Framework requests and responses SHALL use the following CPM Feature tag for service identification:

- The value of 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg' in a P-Preferred-Service or a P-Asserted-Service header field;
- The value of 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg' percent encoded in the Accept-Contact header field as described in [3GPP TS 24.229] section 7.2A.8.2 "Coding of the ICSI" in a g.3gpp.icsi-ref media feature tag of a SIP MESSAGE or SIP INVITE requests and responses.

Additionally, if an Application Server in the network makes use of the CPM Event Reporting Framework to transport events information specific to the application , then it MAY define its own application identifier (IARI) to be used on the CPM request.

As Authenticated Originator's CPM Address (see section 6.1), a request from network to CPM Client SHALL use an identity known to the CPM Client (e.g. the CPM PF's address or a provisioned identity) in the P-Asserted-Identity. The CPM Participating Function MAY set the Authenticated Originator's CPM Address in the P-Served-User header field. To avoid misuse, the CPM Client SHALL only consider requests to be valid requests within the CPM Event Reporting Framework when they are indicated to originate from this known identity.

6.7.2 Supported Event Reporting Scenarios

Several cases are enabled by the CPM Event Reporting Framework, listed herein.

6.7.2.1 CPM Client reported event(s) to CPM Participating Function

The CPM Event Reporting Framework provides several CPM requests and responses mechanisms to transport the event(s) information between a CPM Client that is reporting events that have occurred on the device and the CPM Participating Function. . In such case, the procedures defined in either sections 6.7.3.1 "*Sending One Time CPM Events*" or 6.7.3.3 "*Initiating Bi-directional Session for CPM Events*" SHALL be followed, with the CPM Client acting as the CPM event sender, and the CPM Participating Function acting as the CPM event recipient.

When the CPM Participating Function needs to send responses back to the CPM Client and there is no active session between those two entities allowing to convey such data, the procedures defined in either sections 6.7.3.1 "*Sending One Time CPM Events*" or 6.7.3.3 "*Initiating Bi-directional Session for CPM Events*" SHALL be followed, with the CPM Participating Function acting as the CPM event sender. The CPM Client is acting as the CPM event recipient following the procedures in either section 6.7.3.2 "*Receiving One Time CPM Events*" if the One Time Event procedure was invoked by the CPM event sender, or the procedures in section 6.7.3.4 "*Receiving a Bi-directional Session for CPM Events Invitation*" if the Bi-directional Event Reporting was used by the event sender.

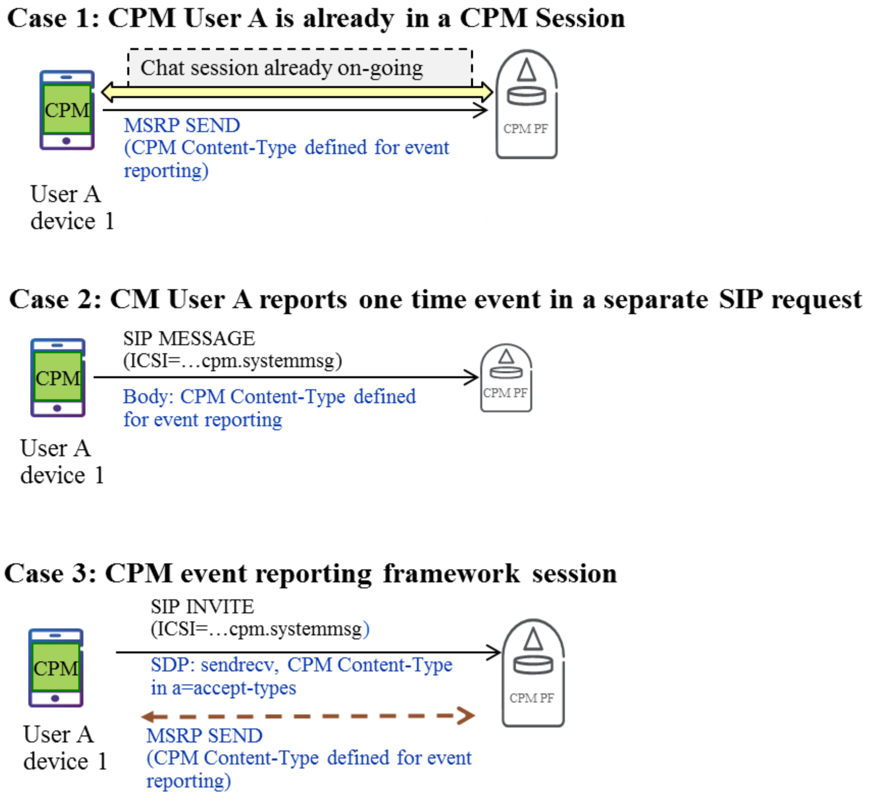
Case 1: CPM User A is already in a CPM Session
Chat session already on-going
CPM
MSRP SEND
(CPM Content-Type defined for event reporting)
CPM PF
User A device 1
Case 2: CM User A reports one time event in a separate SIP request
SIP MESSAGE
(ICSI=…cpm.systemmsg)
CPM
Body: CPM Content-Type defined for event reporting
CPM PF
User A device 1
Case 3: CPM event reporting framework session
SIP INVITE
(ICSI=…cpm.systemmsg)
CPM
SDP: sendrecv, CPM Content-Type in a=accept-types
CPM PF
MSRP SEND
(CPM Content-Type defined for event reporting)
User A device 1

Figure 1 - CPM Event Reporting Framework requests

6.7.2.2 CPM Participating Function requesting event(s) to CPM Client

The CPM Event Reporting Framework allows the CPM Participating Function to request events from a CPM Client (e.g. get the CPM User Activity). In such case, the procedures defined in sections 6.7.3.1 "*Sending One Time CPM Events*" or 6.7.3.3 "*Initiating Bi-directional Session for CPM Events*" SHALL be followed, with the CPM Participating Function acting as the CPM event sender, and the CPM Client acting as the CPM event recipient.

When the CPM Client reports the requested events to the CPM Participating Function and there is no active session between those two entities allowing to convey such data, the procedures defined in sections 6.7.3.1 "*Sending One Time CPM Events*" or 6.7.3.3 "*Initiating Bi-directional Session for CPM Events*" SHALL be followed, with the CPM Client acting as the CPM event sender, and the CPM Participating Function acting as the CPM event recipient.

6.7.2.3 CPM Enabler as a Transport for Other Applications/Enablers

CPM Event Reporting Framework can be used as transport for events and responses of other applications or Enablers, as shown below using OMA EVC Enabler [OMA EVC] as example.

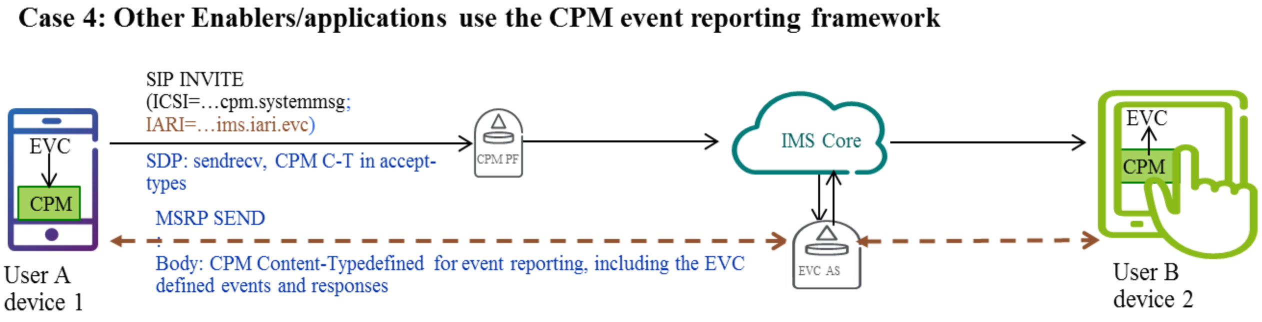
Case 4: Other Enablers/applications use the CPM event reporting framework
SIP INVITE
(ICSI=…cpm.systemmsg;
IARI=…ims.iari.evc)
SDP: sendrecv, CPM C-T in accept-types
MSRP SEND
Body: CPM Content-Typedefined for event reporting, including the EVC defined events and responses
EVC
CPM
User A
device 1
CPM PF
IMS Core
EVC AS
EVC
CPM
User B
device 2

Figure 2 - CPM Event Reporting Framework as transport enabler

6.7.2.4 CPM Event Reporting for Multi-device

The multi-device feature is applicable for the CPM activity event, subject to service provider policies and when the multi-device context is applicable for the CPM User. In this case, the CPM Participating Function MAY inform in real time other CPM Client(s) belonging to the CPM User of the reported event(s). To do so, the following use cases have to be addressed:

When a CPM Chat session allowing the transport of CPM events is already established between the CPM Clients of the served CPM User and the CPM Participating Function, the CPM activity event received from one of the CPM Client MAY, based on service provider policies, be forwarded according to the procedures defined in section 6.7.5.3"*CPM Activity Events*" to the other CPM Clients.

6.7.3 Handling of the CPM Event Reporting requests and responses

6.7.3.1 Sending One Time CPM Events

The CPM event sender SHALL generate a CPM event as described in section 6.7.4 "*CPM Event Reporting Data Definition*". The one time CPM Events SHALL NOT request a processing disposition notification.

If there is already an established MSRP session between the two parties allowing the transport of CPM events, the sending party SHOULD use this session to send the CPM event. Otherwise:

If the size of the CPM event is less than or equal to 1300 bytes, then the CPM event sender SHALL generate a SIP MESSAGE request according to the rules and procedures of [RFC3428]. The CPM event sender:

1. SHALL set the Request-URI and To header field to the address of the intended CPM event recipient;
2. SHALL set the sender's own address in the From header field;
3. SHALL set the P-Preferred-Service header field when the sender is a CPM Client or the P-Asserted-Service header field when the sender is the CPM Participating Function with the value set to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg';

4. SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg' percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI"*;
5. SHALL include a User-Agent header field to indicate the OMA CPM release version as specified in Appendix D "*Release Version in User-agent and Server headers*";
6. SHALL set the body of the SIP MESSAGE request to the generated CPM event;
7. SHALL send the SIP MESSAGE request according to the rules and procedures of the SIP/IP core.

If the size of the CPM event as described in sections 6.7.4 "*CPM Event Reporting Data Definition*" and 6.7.5 "*CPM Defined Events*" is larger than 1300 bytes, then the CPM event sender SHALL generate an initial SIP INVITE request according to the rules and procedures of [RFC3261]. The CPM event sender:

1. SHALL set the Request-URI and To header field to the address of the intended CPM event recipient;
2. SHALL populate the sender's own address in the From header field;
3. SHALL set the the Contact header to a URI identifying its own address;
4. SHALL set the P-Preferred-Service header field when the sender is a CPM Client or the P-Asserted-Service header field when the sender is the CPM Participating Function with the value set to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg';
5. SHALL include a User-Agent header field to indicate the OMA CPM release version as specified in Appendix D "*Release Version in User-agent and Server headers*";
6. SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg' percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI"*;
7. SHALL include the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg' 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 of the Contact header field;
8. If the CPM event sender wants to specify an expiry time for the CPM Standalone Message, SHALL include a Message-Expires header field as defined in Appendix C "*CPM-defined Header fields*";
9. MAY include the option tag 'timer' in the Supported header field;
10. MAY include the Session-Expires header field with the refresher parameter set to "uac" according to the rules and procedures of [RFC4028];
11. SHALL include in the SIP INVITE request a MIME SDP body as a SDP offer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714]. The SDP offer:
    a. SHALL set the SDP directional media attribute to a=sendonly;
    b. SHALL set a=accept-wrapped-types attribute to the CPM Event Reporting Content-Type as defined in section 6.7.4 "*CPM Event Reporting Data Format*";
    c. SHALL set MSRP URI as a=path:MSRP URI;
    d. SHALL set the size as a=file-selector:size:actual message size;
    e. SHALL set the a=setup attribute as "actpass";
    f. SHALL include an SDP 'msrp-cema' attribute in the MSRP media description of the SDP offer;
    g. When negotiation of the maximum MSRP chunk size is supported, SHALL populate the SDP a-line attribute "a=max-chunk-size" containing a value in Kbytes starting from 100 KB;
12. SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.
13. On receiving a SIP 200 "OK" response to the sent SIP INVITE request, the CPM event sender:
    a. SHALL start a SIP session timer using the value received in the Session-Expires header field according to the rules and procedures of [RFC4028] when a Session-Expires header was provided;
    b. SHALL generate and send a SIP ACK request as an acknowledgement of the final response according to the rules and procedures of [RFC3261];
    c. SHALL act as an MSRP client according to [RFC6135] and [RFC6714] with the following clarification:
        i. When in the SDP answer a value is provided, this SHALL be handled as the maximum chunk size;
    d. SHALL generate a CPM event as described in section 6.7.4 "*CPM Event Reporting Data Definition*".

   e. SHALL generate one or more MSRP SEND requests (depending on whether chunking is used) according to the rules and procedures of [RFC4975] and [RFC6714] and the following additional clarifications. The CPM event sender:
      i. SHALL set To-Path header field according to the MSRP URI(s) received in the answer SDP;
      ii. SHALL set the content type as Content-Type= application/vnd.oma.cpm-{event-name}+xml as described in [RFC3862];
      iii. SHALL set the body of the MSRP SEND request(s) to the generated CPM Event.
   f. SHALL send the MSRP SEND requests(s) on the established MSRP connection

14. Once the CPM Event has been successfully transferred via MSRP (i.e. a MSRP 200 response has been received for all MSRP chunks with which the event was transferred), the CPM event sender:
    a. SHALL generate a SIP BYE request related to the previous session according to the rules and procedures of [RFC3261] with Reason code set to SIP protocol and cause=200 (e.g. SIP;cause=200;text="Call completed");
    b. SHALL send the SIP BYE request according to the rules and procedures of SIP/IP core described in [3GPP TS24.229] and SHALL release the media resources;
15. When receiving a MSRP non-200 response to one of the MSRP chunks with which the message is transferred, the CPM event sender SHALL consider that the event has not been processed by the CPM event recipient, and:
    a. SHALL abort the transmission of further MSRP chunks;
    b. SHALL generate a SIP BYE request according to the rules and procedures of [RFC3261] with Reason code set to SIP protocol and cause=480 (e.g. SIP;cause=480;text="Transmission Failed");
    c. SHALL send a SIP BYE request according to the rules and procedures of SIP/IP core described in [3GPP TS24.229] and upon receiving a final response to the SIP BYE request it SHALL release the media resources;

    NOTE: How the CPM event sender handles the error depends on the event type.

6.7.3.2 Receiving One Time CPM Events

Upon receiving a SIP MESSAGE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg' included in the Accept-Contact header field, the CPM event recipient (e.g. an application which makes use of the CPM Event mechanism, CPM Participating Function):

1. SHALL check the incoming request as per section 6.7.3.6 "*Checking CPM Events requests*", and if one of the check fails, or if the request was already forwarded to the SIP/IP Core for further routing then it SHALL not proceed with the rest of the steps;
2. SHALL check if additional IARIs are present in the Accept-Contact header field and if they are supported by the CPM event recipient.
   a. When the CPM event recipient is the CPM Participating Function and:
      i. if there are no additional IARI(s) present in the Accept-Contact header fields, then the request is for a CPM defined event and the CPM Participating Function SHALL continue with the next step; or
      ii. if additional IARI(s) are present but are not used for the CPM functions, then the CPM Participating Function SHALL check the service provider policies for the received IARI(s) and SHALL proceed according to the policies (e.g. MAY send the SIP MESSAGE to the SIP/IP Core for further routing, MAY determine to prohibit certain CPM functions such Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences), and SHALL skip the remaining steps;
   b. For any other CPM event recipients, if they support the received events and the IARI(s) found in Accept-Contact header field(s), then SHOULD continue with the next steps.
3. SHALL check that it supports the type of embedded event, and if not, SHALL return a SIP 406 "Not Acceptable" response, and SHALL not proceed with the rest of the steps;
4. SHALL generate a SIP 200 "OK" response according to rules and procedures of [RFC3428];
5. SHALL send the SIP response according to rules and procedures of the SIP/IP core;
6. SHALL process the CPM event request.

Upon receiving a SIP INVITE request with:

- an Accept-Contact header field set to the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg', and
- an SDP directional media attribute set to a=sendonly,

then the CPM event recipient;

1. SHALL check the incoming request as per section 6.7.3.6 "*Checking CPM Events requests*", and if one of the check fails, SHALL not proceed with the rest of the steps;
2. SHALL check if the accept-wrapped-type attributes of the SDP m line in the SIP INVITE request are acceptable and if not, SHALL reject the request with a SIP 488 "Media Type Not Acceptable Here" response, and not proceed with the rest of the steps.
3. SHALL generate a SIP 200 "OK" response to the received initial SIP INVITE request according to the rules and procedures of [3GPP TS24.229] with the following clarifications:
    a. The CPM event recipient SHALL include an SDP answer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714]. The SDP answer:
        i. SHALL include media line proposing MSRP media parameters;
        ii. SHALL include its own MSRP URI as a=path:MSRP URI;
        iii. SHALL set the SDP directional media attribute to a=recvonly;
        iv. SHALL set a=accept-wrapped-types attribute to the CPM Event Reporting Content-Type as defined in section 6.7.4 "*CPM Event Reporting Data Format*";
        v. SHALL set the a=setup attribute according to [RFC6135];
        vi. SHALL include an SDP 'msrp-cema' attribute in the MSRP media description of the SDP;
        vii. When supporting MSRP chunk size negotiation, SHALL check the SDP a-line attribute "a=max-chunk-size" to determine if the proposed MSRP chunk size is appropriate as per its own configured values based on service provider policies e.g. for the network access type, or the applicable service level agreement;
4. SHALL send the SIP 200 "OK" response according to the rules and procedures of the SIP/IP core.
5. Upon receiving a SIP ACK request, the CPM event recipient:
    a. SHALL handle the SIP ACK request according to the rules and procedures of [RFC3261];
    b. SHALL act as an MSRP endpoint according to [RFC6135] and [RFC6714];

When the CPM event recipient receives an MSRP request, it SHALL follow the rules and procedures defined in [RFC4975] and in [RFC6714]. If an MSRP SEND request indicates the use of chunking, it SHALL wait until all further MSRP SEND requests for the remaining chunks have been received and SHALL reassemble the entire set of MSRP requests into the CPM event. Once the CPM event is received, the CPM event recipient SHALL process it.

Upon receiving a SIP BYE request, the CPM event recipient SHALL respond to the SIP BYE request as described in [3GPP TS24.229] and SHALL release the media resources.

6.7.3.3 Initiating bi-directional Session for CPM Events

When a bi-directional communication between the 2 parties is expected for a sequence of events, such as:

1. the CPM Client expects that additional information is received from the CPM Participating Function to provide additional information or instructions to the CPM Client which reported the event, or
2. the CPM Client expects the CPM Participating Function to inform of the result of the previously reported event(s) which implied an action from the network side (e.g. an <event-cpm-imap > was successfully operated in the CPM Message Store), or
3. the CPM Participating Function requires information from the CPM Client (e.g. CPM User Activity),

then the 2 parties SHALL use a bi-directional MSRP session allowing to exchange CPM events content type.

When no such session is already established, then the CPM event sender SHALL generate an initial SIP INVITE request according to the rules and procedures of [RFC3261]. In this SIP INVITE request, the CPM Event sender:

1. SHALL include the address of the CPM event recipient in the Request-URI and To header field;
2. SHALL populate the sender's own address in the From header field

3. SHALL include an Accept-Contact header field with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg’ , 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;
4. SHALL set the P-Preferred-Service header field when the sender is a CPM Client or the P-Asserted-Service header field when the sender is the CPM Participating Function with the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg’;
5. SHALL include a Contact header field with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg’, 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;
6. SHALL include a User-Agent header field to indicate the OMA CPM release version as specified in Appendix D “*Release Version in User-agent and Server headers*”;
7. MAY include a Supported header field with the option tag 'timer';
8. MAY include the Session-Expires header field with the refresher parameter set to "uac" according to the rules and procedures of [RFC4028];
9. SHALL include a MIME SDP body as a SDP offer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714]. The SDP offer:
   a. SHALL set the SDP directional media attribute to a=sendrecv;
   b. SHALL set a=accept-wrapped-types attribute the CPM Event Reporting Content-Type as defined in section 6.7.4 “*CPM Event Reporting Data Format*“;
   c. SHALL set MSRP URI as a=path:MSRP URI;
   d. SHALL set the a=setup attribute as “actpass”;
   e. SHALL include an SDP 'msrp-cema' attribute in the MSRP media description of the SDP offer;
   f. When negotiation of the maximum MSRP chunk size is supported, SHALL populate the SDP a-line attribute “a=max-chunk-size” containing a value in Kbytes starting from 100 KB;
10. SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

On receipt of the SIP 200 "OK" response to the initial SIP INVITE request the CPM event sender SHALL handle the response according to the rules and procedures of [RFC3261], with the following clarifications:

1. The CPM event sender SHALL generate and send a SIP ACK request as an acknowledgement of the final response according to the rules and procedures of [RFC3261].
2. The CPM Event sender SHALL act as an MSRP endpoint according to [RFC6135] and [RFC6714].
   A. When receiving a MSRP 415, the CPM event sender SHALL stop sending CPM events of the type included in the corresponding MSRP request.

6.7.3.4 Receiving a Bi-directional Session for CPM Events invitation

Upon receiving a SIP INVITE request with:

- an Accept-Contact header field set to the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg’, and
- an SDP directional media attribute set to a=sendrecv,

the CPM event recipient SHALL check:

1. the incoming request as per section 6.7.3.6 “*Checking CPM Events requests*”, and if one of the check fails, SHALL not proceed with the rest of the steps;
2. if the accept- wrapped-type attributes of the SDP m line in the SIP INVITE request are acceptable. If not, the CPM Event recipient SHALL reject the request with a SIP 488 "Media Type Not Acceptable Here" response and SHALL not proceed with the rest of the steps.

Otherwise, the CPM event recipient:

3. SHALL generate a SIP 200 "OK" response towards the intiator of the session;
4. SHALL include a Server header field to indicate the OMA CPM release version as specified in Appendix D “*Release Version in User-agent and Server headers*” depending on whether the sender is a CPM Client or not;
5. SHALL include an SDP answer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714]. The SDP answer:

   a. SHALL include media line proposing MSRP media parameters;
   b. SHALL include its own MSRP URI as a=path:MSRP URI;
   c. SHALL set the SDP directional media attribute to a= sendrecv;
   d. SHALL set a=accept-wrapped-types attribute to the CPM Event Reporting Content-Type as defined in section 6.7.4 "*CPM Event Reporting Data Format*";
   e. SHALL set the a=setup attribute according to [[RFC6135];
   f. SHALL include an SDP 'msrp-cema' attribute in the MSRP media description of the SDP;
   g. When supporting MSRP chunk size negotiation, SHALL check the SDP a-line attribute "a=max-chunk-size" to determine if the proposed MSRP chunk size is appropriate as per its own configured values based on service provider policies e.g. for the network access type, or the applicable service level agreement;

6. SHALL send the SIP 200 "OK" response to the CPM Client according to the rules and procedures of SIP/IP core; and wait for the SIP ACK request from the CPM Client and SHALL propagate it towards the leg on the terminating side.

Upon receiving a SIP ACK request, the CPM event recipient:

1. SHALL handle the SIP ACK request according to the rules and procedures of [RFC3261];
2. SHALL act as an MSRP endpoint according to [RFC6135] and [RFC6714];
   a. For each received CPM event, the CPM event recipient SHALL
      i. check if it supports the type of event, if not, SHALL send a MSRP 415 (which only means that the specifically received CPM event type is not supported and not that negotiated MIME type in the SDP was erroneous), otherwise;
      ii. process it and, when a response to the processing of the event is needed.

6.7.3.5 Closing a Session for CPM Events

The CPM event sender SHOULD close the session when all events have been sent and the needed responses received.

To close a Session for CPM events, the closing entity:

1. SHALL generate a SIP BYE request according to the rules and procedures of [RFC3261], with the Reason Header field as defined in [RFC3326] has a protocol-value set to SIP and a protocol cause=200;
2. SHALL send the SIP BYE request according to the rules and procedures of the SIP/IP core;
3. SHALL release all Media Plane resources corresponding to the session being closed.

6.7.3.6 Checking CPM Events requests

The CPM event recipient SHALL check:

1. that the service is identified as per section 6.7.1*"Service Identification"*, and if not, SHALL return a SIP 403 "Forbidden" response including a SIP Warning header with the warning text set to "127 Service not authorised";
2. that the CPM event sender is allowed to send the event:
   a. When the recipient is the CPM Participating Function, it SHALL check whether the authenticated originator's CPM Address is of a CPM User that is allowed to send a CPM event request (i.e. for the CPM Event Reporting Service identified in the request, identified by the ICSI);
   b. When the recipient is a CPM Client, and the event is a CPM defined event (e.g. IMAP event, an interworking event), it SHALL check that the CPM event sender is the CPM Participating Function;
3. if the CPM event sender is not allowed to send the received event, the CPM event recipient SHALL return a SIP 403 "Forbidden" response including a SIP Warning header with the warning text set to "127 Service not authorised".

The CPM Participating Function SHALL check if additional IARI(s) are present on the request, and if so, then it SHALL check the service provider policies for the received IARI(s) and SHALL proceed according to the policies:

   a. MAY forward the request to the SIP/IP Core for further routing; or
   b. MAY determine if it needs to apply or prohibit any of the CPM features on a CPM requests carrying that IARI(s) (e.g. deferral, interworking, store in the CPM Message Store); or
   c. Any other action indicated by the service provider policies.

6.7.4 CPM Event Reporting Data Definition

CPM defines new Content-Types for the events of the CPM event reporting framework, using the following naming rules:

- "application/vnd.oma.cpm-{event-name}+xml", where the {event-name} is defined by each CPM or non-CPM event specification.

The data of the event(s) reported via the CPM Event Reporting SHALL be included in a CPIM wrapper in the body of the SIP MESSAGE, SIP INVITE or in the body of a MSRP SEND request, then the following clarifications apply for the CPIM wrapper:

1. the From header SHALL be set to the address of the CPM event sender;
2. The To header SHALL be set to the address of the intended CPM event recipient;
3. The MIME Content-Type SHALL be set to: "application/vnd.oma.cpm-{event-name}+xml";
4. When the event requests a response on its processing result, IMDN headers requesting a processing notification as defined in [RFC5438] SHALL be added.
    a. The event receiver SHALL return a IMDN processing notification with the following elements:
        i. SHALL populate the <processed> state element, when the event reported was successfully understood and the requested action was successfully performed;
        ii. SHALL populate the <error> state element, when the event reported was not understood, or it contained erroneous data;
        iii. MAY populate the <forbidden> state element, to communicate to the event sender that the request in the event reported was not allowed to be performed for the requestor.

6.7.4.1 Data Semantics and Validation Constraints

The CPM Event Reporting data MAY include one event. The data structure of the CPM Event Reporting is the following:

1. The top element is <cpm-evfw> and SHALL include:
    a) One child element, containing one of:
        i. a CPM event, as defined in this specification under section 6.7.5 "*CPM Defined Events*", or
        ii. Any other elements from any other namespaces for the purpose of extensibility, containing:
            1) another Enabler's event, defined by the Enabler, or
            2) an application event, defined by that application.

Only the CPM events are described in this specification. However, the CPM Event Reporting Framework provides extensibility points in the data structure that allow other specifications to define their own events and associated data structure.

6.7.5 CPM Defined Events

CPM Enabler defines the following events data:

1) CPM IMAP events;
2) CPM Interworking events for:
    a. CPM User gaining CS connectivity (i.e. being capable to receive SMS deliveries);
    b. Failed SMS delivery via interworking to a CPM User;
3) CPM activity events.

6.7.5.1 CPM IMAP Events

The content type for IMAP events SHALL be: "application/vnd.oma.cpm-event-imap+xml".

The CPM IMAP event provides a data structure that allows the following:

1) Update of IMAP flags (\Deleted and \Seen) for one or more CPM Messages and/or CPM Standalone Media objects, based on CPM User actions that have occurred on the CPM Client (e.g. CPM User has deleted a CPM Message or has displayed a CPM Message).

An <*event-cpm-imap*> element contains the data related to one or more IMAP operations and their related object data. The <*event-cpm-imap*> element has the following attributes:

- Shall have one '*type'* attribute of type String that defines the type of the CPM IMAP event. The attribute can take the following enumerated values: "*flags*";

The <*event-cpm-imap*> element shall have one or more <*operation*> child elements.

The <*operation*> element includes the information associated to the IMAP operation and has the following attributes:

- Shall have one '*name'* attribute is of type String and contains one of the enumerated values for the IMAP flags operations: "*add*";
- Shall have one '*flags'* attribute of type String that contains the one of the enumerated values reflecting the supported IMAP flags:"*\Seen*", "*\Deleted*".

The <*operation*> element has one or more <*object*> child elements.

The <*object*> element is of type String and has the following attributes:

- May have one attribute '*uid*' of type String which includes the value of the CPM Message Store UID of the message object;
- May have one attribute '*folder-path*' of type String which contains the CPM Message Store mailbox storage path of the message object.

The attributes '*uid*' and '*folder-path*' are optional, but the CPM Client shall always include and populate them if they are known for a message object stored in the CPM Message Store.

The <*object*> element may have several child elements. The <*message-id*> child element has the syntax defined in [RFC5438] section 6.3 and it is set to the value of the IMDN <*message-id*> element received in the CPIM header field of the SIP or MSRP request associated with the <*object*>. If the bi-directional SIP INVITE mechanism is used, the <*message-id*> shall be present and populated for each object so the Participating Function can provide the result of the CPM Message Store update for each object using the delivery report IMDN for that <*message-id*> and a processing notification SHALL be requested.

If the CPM Client does not know the message UID value (e.g. message received via SIP by the client and not yet synchronized with the CPM Message Store), then the CPM Client shall include and populate all applicable optional child elements of the <*object*> element, which are used by the CPM Participating Function and CPM Message Store to uniquely identify the corresponding stored message object, as follows:

- one <*conversation-id*> element of type String, set to the value of the Conversation-ID header field received in the SIP request; and,
- one <*contribution-id*> element of type String, set to the value of the Contribution-ID header field received in the SIP request; and,
- one <*message-id*> element with the syntax defined in [RFC5438] section 6.3, set to the value of the IMDN <*message-id*> element received in the CPIM header field of the SIP or MSRP request associated with the message; and,
- for messages belonging to a 1-to-1 conversation, one <other-party> element of type anyURI shall be present and populated with the TEL URI or the SIP URI of the other user.

The events reported by the CPM Client are not rendered to the CPM User, but they SHALL be transparently processed by the CPM Client.

The other CPM Clients SHALL learn of the IMAP event(s) at their next IMAP synchronization.

The CPM Participating Function SHALL generate and send an IMDN processing notification for each of the requests received using the following status values:

a. <processed> to indicate successful result for the requested operation in the event reported; or

b. <error> to indicate that the requested operation was not performed.

6.7.5.2 CPM Interworking Events

The content type for interworking events SHALL be “application/vnd.oma.cpm-event-iw+xml”. Processing disposition notifications SHALL NOT be used for CPM interworking events.

The CPM Participating Function and the CPM Interworking Function communicate via the CPM Interworking event structure, contained in the <*cpm-event-iw*> element. The <cpm-event-iw> element SHALL include one of the following interworking events:

a) if no IMDN, or an IMDN with only “positive-delivery” was requested for the CPM Message, but a SMS Status report containing failure result is received by the CPM Interworking Function for the interworked SMS delivery, then the CPM Interworking Function notifies the CPM Participating Function that the SMS delivery has failed via the CPM Event Reporting Framework, including the <*failed-iw*> event data;

b) when a CPM User, for which a previous SMS delivery has failed, becomes attached or registered and is now available to receive delivery of messages via SMS, the CPM Interworking Function informs the CPM Participation Function via the <*ready-for-sms*> event that a CPM User has now the connectivity necessary to receive SMS.

c) After a successful delivery via interworking (SMS or MMS), the CPM IWF provides the legacy delivery information to the CPM Participating Function in the <*successful-iw-data*> event, for the purpose of storing it in the CPM Message Store. This information is then used by the CPM Client that has received the SMS or MMS delivery, to match the message with its stored version from the CPM Message Store, in order to avoid keeping any duplicate messages on the CPM Client (hence support the CPM Client to only present one copy of the message to the CPM User).

6.7.5.2.1 Interworking Events Structure

The <cpm-event-iw> element has the following structure and SHALL contain only one of the following event elements:

1) zero or one children elements <failed-iw>, populated when the delivery of a message via legacy technology has failed. Each element SHALL have the following children elements:
    a. One <cpm-user-address> element, containing the Authenticated CPM User’s address (e.g. MSISDN) of the CPM User;
    b. One <message-id> element of type String, including the unique identity of the message that has failed delivery;
    c. Other elements, for the purpose of extensibility.
2) zero or one<ready-for-sms> elements, which SHALL include the following children elements:
    a. One <cpm-user-address> element, containing the Authenticated CPM User’s address (e.g. MSISDN) of the CPM User;
    b. Other elements, for the purpose of extensibility.
3) Other elements, for the purpose of extensibility.

The <cpm-event-iw> element SHALL contain at least one of the children elements <failed-iw> or <ready-for-sms> (i.e. it cannot be empty).

6.7.5.3 CPM Activity Events

The content type for activity events SHALL be “application/vnd.oma.cpm-event-activity+xml”.

The CPM activity event provides a data structure that allows the following:

1) Sending a request from the CPM Participating Function to a CPM Client of the time of the last CPM User Activity on a CPM Client or device. In such case, the CPM Participating Function SHALL behave as the CPM event sender as per section 6.7.3.3 “*Initiating Bi-directional Session for CPM Events*”, with the CPM Client as the CPM event recipient.
2) Sending a response from the CPM Client carrying the time of the last CPM User Activity to the requesting CPM Participating Function. In such case, the CPM Client SHALL use the session established as per section 6.7.3.4 “*Receiving a Bi-directional Session for CPM Events Invitation*”.
3) Sending the time of the last CPM User Activity from a CPM Client to the CPM Participating Function, without being requested. In such case, the CPM Client SHALL behave as the CPM event sender as per section 6.7.3.1 “*Sending One Time CPM Events*”, with the CPM Participating Function as the CPM event recipient.

   NOTE: This document only defines the global CPM User Activity structure and procedures. When and how the CPM Client triggers this sort of push of information is out of scope of this specification.

An <*event-cpm-activity*> element contains data allowing to request, and convey, information about the CPM User Activity (i.e. since how long CPM User Activity was detected on a device).

The <*event-cpm-activity*> element is defined as follows:

A. SHALL have one ‘*type’* attribute of type String that defines the type of the event. The attribute can take the following enumerated values:
   i. “*request*”: used when requesting information,
   ii. “*result*”: used when responding to an event of type “*request*”,
   iii. “*push*”: used by the CPM Client to send the CPM User Activity to the CPM Participating Function, without being solicited by the network;
B. SHALL have one ‘id’ attribute of type String populated with a unique value for each event of type “request” or “push”. For the event of type “result”, the ‘id’ attribute shall be populated with the same value of the corresponding received event of type “request”.
C. SHALL have one <global> child element that:
   i. MAY have one ‘time’ attribute of type Integer, which SHALL be present in the event of type “result” or “push”. The ‘time’ attribute SHALL contain the duration in seconds between the last time the CPM Client was aware of any CPM User Activity and the sending of the event.

6.8 CPM Group Session Data Management

The CPM Client and the CPM Controlling function SHALL include the data structure of a request or response operation using the CPM Group Session Data Management schema as defined in Appendix P.1 *"Schema Description"* in the message content of a CPIM message. If the CPIM wrapped message contains multiple message body parts, (e.g. multipart/related, multipart/mixed), then the content types used in the message body parts SHALL be included in the CPIM header field cpm.Payload-Type defined in Appendix C.1.12 “*Payload-Type*”, to provide an indication to the CPM entities or to the SIP/IP Core entities about the content of the payload.

The CPIM message used for the CPM Group Session Data management content SHALL include header fields as follows:

- The IMDN "imdn.Message-ID" header, SHALL be assigned by the entity sending the CPIM wrapper (i.e. the CPM Client or the CPM Controlling Function)CPIM "To", for requests sent by the CPM client it SHALL include the CPM Group Session Identity, otherwise it SHALL be absent,
- CPIM "From", for responses sent by the CPM Controlling Function it SHALL include the CPM Session Identity, otherwise it SHALL be absent,
- CPIM "DateTime", indicating the time the CPIM message was sent by the CPM Client or the CPM Controlling Function

- CPIM cpm.Payload-Type as defined in Appendix C.1.12 "*Payload-Type*", it SHALL be present if the Content-Type of the CPIM wrapped message body is a multipart content type, otherwise it SHALL be absent.

The CPM Group Session Data management enables the CPM Client to request the CPM Controlling Function to manage the meta-data of a CPM Group Session. The Group Session Data management events are initiated by the CPM User request to set, update or remove the subject, the icon or the bulletin of a CPM Group Session. In that case, the CPM Client supporting CPM Group Session Data management SHALL perform the requested meta-data changes to the CPM Group Session conference event information. A CPM Controlling Function that supports the CPM Group Session Data management SHALL process the received requests from the CPM Clients, authorize, perform the authorized operations and inform the Participants of the accepted changes.

The support of CPM Group Session Data management SHALL be negotiated between the CPM Client and the CPM Controlling Function, as defined in sections 7.3 "*CPM Session Handling*" and 9.2 "*CPM Group Session Handling*".

If the CPM User request the CPM Client to perform a CPM Group Session Data management operation while the Long-lived CPM Group Session is inactive, then the CPM Client SHALL re-start the CPM Long-lived Group Session as defined in section 7.3.1.5 *"Re-joining a CPM Long-lived Group Session"* by using the relevant Long-lived CPM Group Session Identity. If the CPM Group Session is active and allows the application of CPM Group Session Data management, then the CPM Client SHALL use the active CPM Group Session to perform the change.

The CPM Client and the CPM Controlling Function communicate via the CPM Group Session Data management structure defined in section 6.8.1 "*Data Structure of the CPM Group Session Data Management Event*" contained in the body of a MSRP SEND request.

The CPM Client SHALL request a change of the CPM Group Session Data of a CPM Group Session by use of the *<group-data>* request element defined below. The CPM Client SHALL assign a unique ID to the request to allow the correlation of the request with a response from the CPM Controlling Function.

To communicate the result of the CPM Client request, the CPM Controlling Function SHALL reply to the CPM Client by use of the <group-data> response element in the active CPM Group Session. The successful result of processing the request is conveyed to the Participants (including the initiator's CPM Client) via the update of Participant Information of the CPM Group Session as defined in section 9.2.14.3 "*Sending Participant Information Notifications*".

6.8.1 Data Structure of the CPM Group Session Data Management

The CPM Group Session Data management SHALL have the following data structure:

1) A top element *<cpm-event-group-management>* with the following attributes and children elements:
    a) SHALL have one attribute "id" of type string containing a unique request ID assigned by the CPM Client for the CPM Group Session DataGroup Session Data management request. The same unique request ID SHALL be returned by the CPM Controlling Function to the CPM Client in the response containing the result of this request.
    b) MAY have a child element *<group-data >* element, with either one of the following child elements:
        A. a <request> element which :
            1) Group Session Data SHALL have one <target> element of type String that can take one of the following values:
                a. "subject", which indicates that the request of the CPM Client targets the <subject> element and the <subject-ext> element data extension to the conference event package, as defined in Appendix P;
                b. "icon", which indicates that the request of the CPM Client targets the <icon> element of the OMA CPM extension to the conference event package, as defined in Appendix P.
                c. "bulletin", which indicates that the request of the CPM Client targets the <bulletin> element extension to the conference event package, as defined in Appendix P.
                d. "role", which indicates that the request of the CPM Client targets the role of a CPM user, as defined in Annex Q.1.2 *"CPM User Roles"*.
            2) SHALL have one <action> element of type String with the following enumerated values:

a. "set" indicates that the request of the CPM Client contains a new value for the element indicated in the “target” attribute;

b. "delete" indicates that the request of the CPM Client requests the CPM Controlling Function to delete the values of elements indicated in the “target” attribute;

c. "move" indicates that the request of the CPM Client moves a value of the element indicated in the "target" attribute. The "move" action SHALL be used by the CPM Client only for the target "role" to move a single role assigned to its CPM User to another CPM User.

3) MAY have one <data> child element containing the new value of the element indicated via the <target> element. The <data> child element SHALL be present if the <action> attribute has the value "set". The <data> SHALL have one of the following child elements:

a. a <subject> element of type String containing the subject to be assigned to the CPM Long-lived Group session. The <subject> element SHALL be present if the value of the <target> attribute is set to "subject" and the value of the <action> attribute is set to "set".

b. a <icon> element containing either one of the following child elements:

i. <icon-uri> element of type anyURI containing the URI pointing to the image. The procedure for the CPM Client to determine a URI for the icon image is out of scope of this specification.

ii. <file-info> element of type String containing a content-id (CID) pointing to an additional body of the MSRP SEND message containing the image file.

1. The second CPIM Content-Type in the MSRP SEND message SHALL be set to MIME multipart/related, and a CPIM cpm.Payload-Type header field SHALL be added including a listing with both the CPM Group Session Data management content type and the content type of the file;

2. A CPM Client that supports the <file-info> element and the population of the binary file that includes the icon, in the CPM Group Session Data management operations, SHALL populate in the SDP a=accept-wrapped-types also the supported content types for the images that can be used as CPM Group Session Data icons.

c. a <bulletin> element of type String containing the bulletin to be assigned to the CPM Long-lived Group session. The <bulletin> element SHALL be present if the value of the <target> attribute is set to "bulletin" and the value of the <action> attribute is set to "set";

d. <role> element containing both of the following child elements:

iii. <user-role> element of type string containing a single role value as defined in Appendix Q.1.2 "CPM User Roles";

iv. <participant> element of type anyURI containing the communication address of the CPM User targeting the request for change of the role.

B. a <response> element of type String, which contains the following children element:

1) SHALL have one <response-code> element of type Integer containing the value of the result following the SIP response codes rules (e.g. “200” value of result is successful, or a “403” value if the change was forbidden due to authorization reasons).

7. Procedures at CPM Client

A CPM Client provides the CPM User with an interface to access the functionalities of the CPM Enabler.

Before a CPM Client can provide CPM functionalities to the CPM User it has to register to the SIP/IP core as described in section 7.1 "*Registering at the SIP/IP Core*".

A CPM Client can initiate new communications in a number of ways, as follows:

- When a CPM Client needs to send a Pager Mode CPM Standalone Message, it SHALL follow the procedures described in section 7.2.1.1 "*Sending a Pager Mode CPM Standalone Message*".
- When a CPM Client needs to send a Large Message Mode CPM Standalone Message, it SHALL follow the procedures described in section 7.2.1.2 "*Sending a Large Message Mode CPM Standalone Message*".
- When a CPM Client needs to initiate a new CPM Session or re-join a previous CPM Session, it SHALL follow one of the procedures described in section 7.3.1 "*Initiating New CPM Sessions*", depending on what kind of CPM Session is to be initiated.
- When a CPM Client needs to initiate a new CPM File Transfer, it SHALL follow the procedures defined in section 7.4.1 "*CPM File Transfer Session Initiation*".
- When a CPM Client needs to handle Deferred CPM Message(s), it SHALL follow the procedures defined in section 7.2.3.2 "*Handling Deferred CPM Message(s)*".
- When a CPM Client needs to handle Deferred CPM 1-1 Session Message(s) delivery, it SHALL follow the procedures defined in section 7.3.11 "*Handling Deferred CPM 1-1 Session Message Delivery*".
- When a CPM Client needs to handle Deferred CPM Group Session Message(s) delivery, it SHALL follow the procedures defined in section 7.3.12 "*Handling Deferred CPM Group Session Message Delivery*".
- When a CPM Client needs to send a CPM event (as defined in section 6.7), it SHALL follow the procedures described in section 6.7.3.1 "*Sending One Time CPM Events*" or 6.7.3.3 "*Initiating Bi-directional session for CPM Events*".

A CPM Client handles incoming CPM communications in the following manners:

- Upon receiving a SIP MESSAGE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg' or with CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred' (defined in Appendix H) percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI,* included in the Accept-Contact header field, the CPM Client SHALL handle this SIP MESSAGE request as described in section 7.2.2.1 "*Receiving a Pager Mode CPM Standalone Message*".
- Upon receiving a SIP MESSAGE request with the CPM Feature Tag '*urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg*' (defined in Appendix H) included in the Accept-Contact header field, the CPM Client SHALL handle this SIP MESSAGE request as described in section 6.7.3.2 "*Receiving One Time CPM Event*".
- Upon receiving a SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg' or with CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred' (defined in Appendix H) percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI,* included in the Accept-Contact header field, the CPM Client SHALL handle this SIP INVITE request as described in section 7.2.2.2 "*Receiving a Large Message Mode CPM Standalone Message*". Upon receiving a SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' or with CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred' (defined in Appendix H) percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI,* included in the Accept-Contact header field, the CPM Client SHALL handle this SIP INVITE request as described in section 7.3.2 "*Receiving a CPM Session Invitation*", or section 7.3.12 "*Handling Deferred CPM Group Session Message Delivery* " or section 7.3.11 "*Handling Deferred CPM 1-1 Session Message Delivery*",
- Upon receiving a SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer' (defined in Appendix H) included in the Accept-Contact header field, the CPM Client SHALL handle this SIP INVITE request as described in section 7.4.2 "*Receiving a CPM File Transfer Request*",

- Upon receiving a SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg' (defined in Appendix H) included in the Accept-Contact header field, the CPM Client SHALL handle this SIP INVITE request as described:
  - in section 6.7.3.2 "*Receiving One Time CPM Events*" when the SDP directional media attribute is set to a=sendonly, or
  - in section 6.7.3.4 "*Receiving a Bi-directional Session for CPM Event Invitation*" when the SDP directional media attribute is set to a= sendrecv.

The CPM Client SHALL include a User-Agent header field to indicate the OMA CPM release version of the CPM Client as specified in Appendix D "*Release Version in User-agent and Server headers*".

7.1 Registering at the SIP/IP Core

The CPM Client's CPM service registration or re-registration to the SIP/IP core SHALL be made according to the rules and procedures of [3GPP TS.24.229] and the SIP/IP core.

When registering or re-registering for the CPM service, the CPM Client:

1. SHALL generate a SIP REGISTER request according to the rules and procedures of [3GPP TS.24.229];
2. SHALL include the CPM Feature Tags of the supported features in the Contact header field as described in Appendix H "*CPM Feature Tags*";
3. SHALL include a +sip.instance header field parameter in each contact included in the SIP REGISTER,with the instance identifier value (called instance ID in [RFC5627]) of the CPM Client in the Contact header field according to [RFC5626]. The value of the sip.instance SHOULD be set according to [3GPP TS.24.229];
4. SHALL indicate CPM Client user agent capabilities in the Contact header field according to [RFC3840];
5. SHALL include a User-Agent header field to indicate the OMA CPM release version of the CPM Client as specified in Appendix D "*Release Version in User-agent and Server headers*";
6. SHALL indicate the name of the CPM Client, as provided by the CPM User, in the display-name part of the Contact header field as provided by the CPM User;
7. MAY include a Supported header field with the option tag 'gruu' according to [RFC5627];
8. MAY include a Require header field with the option tag 'gruu' according to [RFC5627];
9. SHALL send the SIP REGISTER request towards the SIP/IP core according to the rules and procedures of the SIP/IP core.

To maintain the SIP registration for the CPM service active, the CPM Client SHALL use the re-registration procedure as specified in [RFC3261].

When the CPM service is no longer required, the CPM Client SHALL send a SIP REGISTER request without CPM Feature Tags of the supported features in the Contact header field as described in Appendix H "*CPM Feature Tags*"; according to the rules and procedures of [RFC3261] terminating the existing CPM registration. If the CPM Client also needs to deregister from the SIP/IP core, it SHALL send a SIP REGISTER request with an Expires header field set to 0, as defined in [RFC3261]

If device settings are to pull Deferred CPM Messages, the CPM Client SHALL subscribe to the "deferred-messages" event package as described in section 7.2.3.1 "*Subscribe to Deferred CPM Message Info*".

7.2 CPM Standalone Message Handling

CPM Clients MAY support different text and media types (including SMIL) as defined in [3GPP TS 26.141].

7.2.1 Sending CPM Standalone Messages

7.2.1.1 Sending a Pager Mode CPM Standalone Message

When the CPM Client wants to send a CPM Standalone Message and the CPM Standalone Message is small enough (i.e. less than or equal to 1300 bytes) to be sent as a Pager Mode CPM Standalone Message, the CPM Client SHALL generate a SIP MESSAGE request according to the rules and procedures of [RFC3428]. The CPM Client:

1. SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg' percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 "*Coding of the ICSI*";

2. MAY include one of the registered CPM Addresses of the CPM User as described in section 6.1 “*Authenticated Originator’s CPM Address*” as authenticated originator's CPM Address;
3. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag as follows:
    a. If the Pager Mode CPM Standalone Message is sent to one recipient, the ICSI value SHALL be set to ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg’;
    b. Otherwise, if the Pager Mode CPM Standalone Message is sent to a group of recipients, the ICSI value SHALL be set to ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.group’
4. SHALL include a Reply-To header field if the CPM User requests a different reply address from the address included in the From header field;
5. SHALL include a User-Agent header field to indicate the OMA CPM release version of the CPM Client as specified in Appendix D “*Release Version in User-agent and Server headers*”;
6. If anonymity is requested, SHALL include value "id" in a Privacy header field according to the rules and procedures of [RFC3325];
7. If the CPM User or CPM Client wants to specify an expiry time for the CPM Standalone Message, SHALL include an Expires header field as defined in [RFC3261];
8. If the CPM Standalone Message is to be sent to one CPM user or to one non-CPM Principal, the CPM Client SHALL set the Request-URI of the SIP MESSAGE request as follows:
    a. In case of a new CPM Standalone Message to the address of the target CPM User or the target non-CPM Principal;
    b. In case of a reply and a Reply-To header field is included in the received CPM Standalone Message for which the CPM Standalone Message is a reply to, to the address in that Reply-To header field, unless a new address is provided by the CPM User;
    c. In case of a reply and no Reply-To header field is included in the received CPM Standalone Message for which the CPM Standalone Message is a reply to, to the address in the header field containing the authenticated originator's CPM Address, unless a new address is provided by the CPM User.
9. If the CPM Standalone Message is to be sent to a CPM Ad-hoc Group, then the CPM Client:
    a. SHALL set the Request-URI to the address of the CPM Controlling Function provisioned to the CPM Client;
    b. SHALL include a MIME resource-list body with the target addresses of the CPM Users and non-CPM Principals as specified in [RFC5365];
    c. In case of a reply and a Reply-To header field is included in the received CPM Standalone Message for which the CPM Standalone Message is a reply to, SHALL insert to the MIME resource-list body at least the address of that Reply-To header field, unless a new address is provided by the CPM User;
    d. In case of a reply and no Reply-To header field is included in the received CPM Standalone Message for which the CPM Standalone Message is a reply to, SHALL insert to the MIME resource-list body at least the address in the header field containing the authenticated originator's CPM Address, unless a new address is provided by the CPM User.
10. If the CPM Standalone Message is to be sent to a CPM Pre-defined Group, then the CPM Client SHALL set the Request-URI to the address of the target CPM Pre-defined Group;
11. If the CPM Standalone Message starts a new CPM Conversation, the CPM Client SHALL include a Conversation-ID header field set to a newly generated value as specified in Appendix C “*CPM-defined Header fields*”, Otherwise, the CPM Client SHALL include the Conversation-ID of the existing CPM Conversation;
12. The CPM Client SHALL include a Contribution-ID header field with a newly generated Contribution-ID value as specified in Appendix C “*CPM-defined Header fields*”;
13. If the CPM Standalone Message is in reply to a previously received CPM Standalone Message, CPM File Transfer or CPM Session Invitation, the CPM Client SHALL include the InReplyTo-Contribution-ID header field with the value of Contribution-ID in the previously received CPM Standalone Message, CPM File Transfer or CPM Session Invitation;
14. The CPM Client SHALL generate a CPIM message wrapper as defined in section 7.2.1.3. “*Generate a CPM Standalone Message*”;
15. The CPM Client SHALL set the body of the SIP MESSAGE request to the generated CPIM message wrapper;

16. The CPM Client SHALL send the SIP MESSAGE request according to the rules and procedures of the SIP/IP core.
17. If the CPM Message was stored in the CPM Message Store by the CPM Participating Function, the CPM Client SHALL retrieve from the SIP response 200 “OK” also the Message-UID header field containing the UID value of the CPM Message stored in the CPM Message Store. This value is used further by the CPM Client in the message synchronization process with CPM Message Store via IMAP, to correlate with the CPM Messages handled via the SIP channel.
18. If a Message-UID header field is not present in the SIP 200 OK, or the SIP 200 OK fails to arrive at the CPM Client, the CPM Client SHALL identify when message duplicates exist locally in the client (e.g. received via the SIP delivery and also received via IMAP synchronization) based on:
    a. the IMAP folder in which the message is stored (i.e. folder SHALL match the identity of the other party in a 1-1 CPM Standalone Message and the Conversation-ID for the CPM Group Standalone Message) and,
    b. the CPM SIP headers fields of a message delivered via SIP match the values of the MIME headers of message objects received during the IMAP synchronization, as described in the table below:

CPM SIP Header field or CPIM body header field, as defined in Appendix C	**CPM MIME Header (IMAP),** **as defined in Appendix C of [OMA-CPM_TS_MessageStorage]**	**Matching**
Conversation-ID	Conversation-ID	Yes
Contribution-ID	Contribution-ID	Yes
InReplyTo-Contribution-ID (if present)	InReplyTo-Contribution-ID	Yes, if present
IMDN Message-ID header value from the CPIM body as defined in [RFC5438] (if applicable)	IMDN-Message-ID	Yes, for all CPM Messages and CPM File Transfers with a CPIM body.
cpm.Payload-Type	Payload-Type	Not an absolute matching criteria. It is possible that one Content-Type in the payload was supported by one CPM Client of the CPM User) e.g. the one the message was originally delivered to), but is not supported by the CPM Client that obtained the CPM Message via synchronization with the Message Store.

Table 2: Matching criteria for messages by the CPM Client

7.2.1.2 Sending a Large Message Mode CPM Standalone Message

When the CPM Client wants to send a CPM Standalone Message and the CPM Standalone Message is too large (i.e. larger than 1300 bytes) to be sent as a Pager Mode CPM Standalone Message, the CPM Client SHALL generate an initial SIP INVITE request according to the rules and procedures of [RFC3261]. The CPM Client:

1. SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg' percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI"*;
2. SHALL include the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg' 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 of the Contact header field;
3. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag as follows:
    a. If the Large Message Mode CPM Standalone Message is sent to one recipient, the ICSI value SHALL be set to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg';
    b. Otherwise, if the Large Message Mode CPM Standalone Message is sent to a group of recipients, the ICSI value SHALL be set to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group'
4. MAY include one of the registered CPM Addresses of the CPM User as described in section 6.1 "*Authenticated Originator's CPM Address*" as authenticated originator's CPM Address;
5. SHALL include a Reply-To header field if the CPM User requests a different reply address;
6. SHALL include a User-Agent header field to indicate the OMA CPM release version of the CPM Client as specified in Appendix D "*Release Version in User-agent and Server headers*";
7. If anonymity is requested, SHALL include value "id" in a Privacy header field according to the rules and procedures of [RFC3325];
8. If the CPM User or CPM Client wants to specify an expiry time for the CPM Standalone Message, SHALL include a Message-Expires header field as defined in Appendix C "*CPM-defined Header fields*";
9. If the Large Message Mode CPM Standalone Message is to be sent to one CPM user or to one non-CPM Principal, the CPM Client SHALL set the Request-URI of the SIP INVITE request as follows:
    a. In the case of a new CPM Standalone Message, to the address of the target CPM User or the target non-CPM Principal;
    b. In case of a reply and a Reply-To header field is included in the received CPM Standalone Message, CPM File Transfer or CPM Session Invitation for which the CPM Standalone Message is a reply to, to the address of that Reply-To header field, unless a new address is provided by the CPM User;
    c. In case of a reply and no Reply-To header field is included in the received CPM Standalone Message, CPM File Transfer or CPM Session Invitation for which the CPM Standalone Message is a reply to, to the address of the header field containing the authenticated originator's CPM Address, unless a new address is provided by the CPM User.
10. If the Large Message Mode CPM Standalone Message is to be sent to a CPM Ad-hoc Group, then the CPM Client:
    a. SHALL set the Request-URI to the address of the CPM Controlling Function provisioned to the CPM Client;
    b. SHALL include a MIME resource-list body with the target addresses of the CPM Users and non-CPM Principals as specified in [RFC5364];
    c. In case of a reply and a Reply-To header field is included in the received CPM Standalone Message for which the CPM Standalone Message is a reply to, SHALL insert to the MIME resource-list body at least the address of that Reply-To header field, unless a new address is provided by the CPM User; In case of a reply and no Reply-To header field is included in the received CPM Standalone Message, CPM File Transfer or CPM Session Invitation for which the CPM Standalone Message is a reply to, to the address in the header field containing the authenticated originator's CPM Address of the received CPM Standalone Message, unless a new address is provided by the CPM User.
11. If the Large Message Mode CPM Standalone Message is to be sent to a CPM Pre-defined Group, then the CPM Client SHALL set the Request-URI to the address of the target CPM Pre-defined Group;
12. SHALL include the option tag 'timer' in the Supported header field;

13. SHALL NOT include the Session-Expires header field with the refresher parameter set to "uac" according to the rules and procedures of [RFC4028];
14. SHALL NOT include the Min-SE header field according to the rules and procedures of [RFC4028];
15. If the Large Message Mode CPM Standalone Message starts a new CPM Conversation, the CPM Client SHALL include a Conversation-ID header field set to a newly generated value as specified in Appendix C "*CPM-defined Header fields"*. Otherwise, the CPM Client SHALL include the Conversation-ID of the existing CPM Conversation.
16. The CPM Client SHALL include a Contribution-ID header field with a newly generated Contribution-ID value as specified in Appendix C "*CPM-defined Header fields"*;
17. If the Large Message Mode CPM Standalone Message is in reply to a previously received CPM Standalone Message , CPM File Transfer or CPM Session Invitation, the CPM Client SHALL include the InReplyTo-Contribution-ID header field with the value of Contribution-ID in the previously received CPM Standalone Message, CPM File Transfer or CPM Session Invitation as specified in Appendix C "*CPM-defined Header fields"*;
18. SHALL include in the SIP INVITE request a MIME SDP body as a SDP offer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714] with the following clarification:
    a. SHALL set the SDP directional media attribute to a=sendonly;
    b. SHALL set the content type as a=accept-types:message/cpim;
    c. SHALL set the sub-content type as a=accept-wrapped-types:*;
    d. SHALL set MSRP URI as a=path:MSRP URI;
    e. SHALL set the size as a=file-selector:size:actual message size;
    f. SHALL set the a=setup attribute as "actpass";
    g. SHALL include an SDP 'msrp-cema' attribute in the MSRP media description of the SDP offer.

       NOTE: The value '*' is for example and implementations shall set this value to the MIME sub-media types of the CPIM.
19. The CPM Client SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

When the CPM Client needs to cancel the message delivery and if the CPM Client has not yet received a final SIP response for the SIP INVITE request, the CPM Client SHALL send a SIP CANCEL request according to the rules and procedures of [RFC3261].

On receiving a SIP 200 "OK" response to the sent SIP INVITE request, the CPM Client:

A. SHALL start a SIP session timer using the value received in the Session-Expires header field according to the rules and procedures of [RFC4028];
B. SHALL generate and send a SIP ACK request as an acknowledgement of the final response according to the rules and procedures of [RFC3261];
C. SHALL act as an MSRP client according to [RFC6135];
D. SHALL act as an active endpoint to open the transport connection according to [RFC6135];
E. SHALL establish the MSRP connection according to the MSRP connection parameters (c/m-line address information) in the SDP answer received in the SIP 200 "OK" response according to [RFC6714];
F. SHALL send an empty MSRP SEND request to bind the MSRP connection to the MSRP session from the perspective of the passive endpoint according to the rules and procedures of [RFC4975] and [RFC6135];
G. SHALL generate a CPIM message wrapper as defined in section 7.2.1.3 "*Generate a CPM Standalone Message*".
H. SHALL generate one or more MSRP SEND requests (depending on whether chunking is used) according to the rules and procedures of [RFC4975] and [RFC6714], taking into account the maximum chunk size negotiated according to section 5.2.1, if any, and the following additional clarifications. The CPM Client:
    1. SHALL set To-Path header field according to the MSRP URI(s) received in the answer SDP;
    2. SHALL set the content type as Content-Type = message/cpim as described in [RFC3862];
    3. SHALL set a second Content-Type header field to the content type of the payload inside the CPIM wrapper. If the CPIM wrapped message body includes a multipart MIME (e.g. multipart/related, multipart/mixed), then:
        i. SHALL add a CPIM cpm.Payload-Type header field, set to the list of Content-Types included in the payload of the body of the message.

I. SHALL send the MSRP SEND requests(s) on the established MSRP connection.

Upon receiving other non-2XX final responses to the sent SIP INVITE request the CPM Client:

A. SHALL generate and send a SIP ACK request as an acknowledgement of the final response according to the rules and procedures of [RFC3261];
B. , SHALL indicate to the CPM User that the transfer of the CPM Standalone Message has failed unless the CPM Client supports a seamless fall-back to non-CPM technologies for the transfer of the message.

Once the CPM Standalone Message has been successfully transferred via MSRP (i.e. a MSRP 200 response has been received for all MSRP chunks with which the message was transferred), the CPM Client:

1) SHALL generate a SIP BYE request according to the rules and procedures of [RFC3261] with Reason code set to SIP protocol and cause=200 (e.g. SIP;cause=200;text="Call completed");
2) SHALL send a SIP BYE request according to the rules and procedures of SIP/IP core described in [3GPP TS24.229] and upon receiving a final response to the SIP BYE request it SHALL release the media resources;
3) If the CPM Message was stored in the CPM Message Store by the CPM Participating Function, the CPM Client SHALL retrieve from the SIP response 200 “OK” to the SIP BYE request, also the Message-UID header field containing the UID value of the CPM Message stored in the CPM Message Store. This value is used further by the CPM Client in the message synchronization process with CPM Message Store via IMAP, for correlation with the CPM Messages handled via the SIP channel.
4) If a Message-UID header field is not present in the SIP 200 “OK” response to the SIP BYE request, or the SIP 200 “OK” response fails to arrive at the CPM Client, the CPM Client SHALL identify when message duplicates exist locally in the client (e.g. received via the SIP delivery and also received via IMAP synchronization) based on:
    a. the IMAP folder in which the message is stored (i.e. folder SHALL match the identity of the other party in a 1-1 CPM Standalone Message and the Conversation-ID for the CPM Group Standalone Message) and,
    b. the CPM SIP headers fields of a message delivered via SIP match the values of the MIME headers of message objects received during the IMAP synchronization, as described in the table 1 section 7.2.2.1 “*Sending a Pager Mode CPM Standalone Message and SIP IMDNs*”.

When receiving a MSRP non-200 response to one of the MSRP chunks with which the message is transferred, the CPM Client:

1. SHALL abort the transmission of further MSRP chunks;
2. SHALL generate a SIP BYE request according to the rules and procedures of [RFC3261] with Reason code set to SIP protocol and cause=480 (e.g. SIP;cause=480;text="Transmission Failed");
3. SHALL send a SIP BYE request according to the rules and procedures of SIP/IP core described in [3GPP TS24.229] and upon receiving a final response to the SIP BYE request it SHALL release the media resources;
4. SHALL unless the client supports a seamless fall-back to non-CPM technologies for the transfer of the message indicate to the CPM User that the transfer of the CPM Standalone Message has failed.

7.2.1.3 Generate a CPM Standalone Message

In order to generate a CPM Standalone Message, a CPM Client SHALL generate a CPIM message as defined in [RFC3862], with the following clarifications:

1. SHALL set the From header field of the CPIM message to the address of the sending CPM User, as described in section 6.1.1 "*Identifying the sending device in SIP requests and responses*";
2. SHALL set the To header field of the CPIM message as follows:
    a. If the CPM Standalone Message is to be sent to one CPM User or to one non-CPM Principal, the CPM Client SHALL set the To header field of the CPIM message to the address of the recipient;
    b. If the CPM Standalone Message is to be sent to a CPM Ad-hoc Group, the CPM Client SHALL set the To header field of the CPIM message to the address of the CPM Controlling Function provisioned to the CPM Client;
    c. If the CPM Standalone Message is to be sent to a CPM Pre-defined Group, the CPM Client SHALL set the To header field of the CPIM message to the address of the target CPM Pre-defined Group;
3. SHALL set a second Content-Type header field to the content type of the payload that it placed inside the CPIM wrapper. If this CPIM wrapped message body includes a multipart MIME (e.g. multipart/related, multipart/mixed), then:
    i. SHALL add a CPIM cpm.Payload-Type header field, set to the list of Content-Types included in the payload of the body of the message.
4. If the CPM user wishes to obtain the disposition-state of the CPM Standalone Message (e.g. whether the message is delivered or interworked to the recipients or not, whether the recipients read the message), the CPM Client SHALL set the disposition notifications requested (delivered, interworked and/or display notifications) according to the rules and procedures described in section 5.4 "*Disposition Notifications*".
5. If the size of the message is over 1300 bytes, the CPM Client SHALL perform the procedures specified in 7.2.1.2 "*Sending a Large Message*". Otherwise, the CPM Client SHALL perform the procedures specified in 7.2.1.1 "*Sending a Pager Mode CPM Standalone Message*".

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

When a CPM Client generates a SIP MESSAGE or MSRP SEND request with references to stored data (e.g. Media Objects, CPM Standalone Messages, CPM Session Histories or CPM Conversation History) in the Message Storage Server, the CPM Client:

1. SHALL obtain the URL to the stored data from the Message Storage Client;

   NOTE: The Message Storage Client obtains the URL from the Message Storage Server.

2. SHALL include a Content-Type header field as specified in [RFC4483] (i.e. type set to "message/external-body", access-type set to "URL", and the URL obtained in step 1 in the URL parameter). When including a URL, the URL SHALL be modified to include the extension "?action=fetch";

   NOTE: If there are other Content-Type header fields within the SIP MESSAGE request, the CPM Client shall include the Content-Type header field "multipart/mixed" with the proper boundary value to distinguish the URL part and existing parts as described in [RFC2046].

3. If the size of the message is over 1300 bytes, the CPM Client SHALL perform the procedures specified in 7.2.1.2 "*Sending a Large Message*". Otherwise, the CPM Client SHALL perform the procedures specified in 7.2.1.1 "*Sending a Pager Mode CPM Standalone Message*".

7.2.2 Receiving CPM Standalone Messages

7.2.2.1 Receiving a Pager Mode CPM Standalone Message and SIP IMDNs

Upon receiving a SIP MESSAGE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg' included in the Accept-Contact header field corresponding to Pager Mode CPM Standalone Message, or with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred' included in the Accept-Contact header field corresponding to a Deferred CPM Message, or with one of the CPM Feature Tags 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg', or'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session', or 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer' included in the Accept-Contact header field corresponding to a SIP IMDN received in the SIP MESSAGE, the CPM Client:

1. MAY reject the SIP MESSAGE request if it determines that there are not enough resources to handle the SIP MESSAGE request and return a SIP 480 “Temporarily Unavailable” response;
2. MAY reject the message if none of the Content-Types listed in the CPIM cpm.Payload-Type header field is supported by the CPM Client with a SIP 488 “Not acceptable here” response.

Otherwise, the CPM Client:

1. SHALL generate a SIP 200 “OK” response according to rules and procedures of [RFC3428];
2. SHALL include a Server header field to indicate the OMA CPM release version of the CPM Client as specified in Appendix D “*Release Version in User-agent and Server headers*”.
3. MAY include one of the registered CPM Addresses of the CPM User as described in section 6.1 “*Authenticated Originator’s CPM Address*” as authenticated recipient 's CPM Address;
4. SHALL include all of the Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID header fields received in the SIP MESSAGE request;
5. SHALL send the SIP response according to rules and procedures of the SIP/IP core;
6. SHALL handle the CPIM message of the received Pager Mode CPM Standalone Message and SIP IMDNs as described in section 7.2.2.3 “*CPIM Handling of Received CPM Standalone Messages and SIP IMDNs*”.
7. If the CPM Message was stored in the CPM Message Store by the CPM Participating Function on the terminating side, the CPM Client SHALL retrieve from the SIP MESSAGE request the Message-UID header field containing the UID value of the CPM Message stored in the CPM Message Store. This value is used further by the CPM Client in the message synchronization process with CPM Message Store via IMAP, for correlation with the CPM Messages handled via the SIP channel. If a Message-UID header field is not present in the SIP MESSAGE, the CPM Client SHALL identify when message duplicates exist locally in the client (e.g. received via the SIP delivery and also received via IMAP synchronization) based on:
    a. the IMAP folder in which the message is stored (i.e. folder SHALL match the identity of the other party in a 1-1 CPM Standalone Message and the Conversation-ID for the CPM Group Standalone Message) and,
    b. the CPM SIP headers fields of a message delivered via SIP match the values of the MIME headers of message objects received during the IMAP synchronization, as described in the table 2 section 7.2.1.1 “*Sending a Pager Mode CPM Standalone Message*”.
8. If a Message-UID header field as defined in Appendix C “*CPM-defined SIP Headers*” is included, SHALL store the UID value in conjunction with the message if storing the received CPM Standalone Message in the local storage.

7.2.2.2 Receiving a Large Message Mode CPM Standalone Message

Upon receiving a SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg’ included in the Accept-Contact header field corresponding to Large Message Mode CPM Standalone Message, or upon receiving a SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred’ included in the Accept-Contact header field corresponding to Deferred Large Message Mode CPM Standalone Message, the CPM Client MAY reject the SIP INVITE request if it determines that there are not enough resources to handle the SIP INVITE request and return a SIP 480 “Temporarily Unavailable” response;

Otherwise, the CPM Client:

1. SHALL check if the accept-type and accept-wrapped-types attributes of the SDP m line in the SIP INVITE request are acceptable to the CPM Client and if not, reject the request with a SIP 488 "Media Type Not Acceptable Here" response.

   Otherwise, continue with the rest of the steps;
2. SHALL generate a SIP 200 “OK” response to the received initial SIP INVITE request according to the rules and procedures of [3GPP TS24.229] with the following clarifications:
    a. The CPM Client SHALL include a Server header field to indicate the OMA CPM release version of the CPM Client as specified in Appendix D “*Release Version in User-agent and Server headers*”
    b. The CPM Client MAY include one of the registered CPM Addresses of the CPM User as described in section 6.1 “*Authenticated Originator’s CPM Address*” as authenticated recipient 's CPM Address;
    c. The CPM Client SHALL include an SDP answer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714] with the following clarifications:
        i. The CPM Client SHALL include media line proposing MSRP media parameters;

ii. The CPM Client SHALL include its own MSRP URI as a=path:MSRP URI;

iii. The CPM Client SHALL set the SDP directional media attribute to a=recvonly;

iv. The CPM Client SHALL set the content type as a=accept-types:message/cpim;

v. The CPM Client SHALL set the sub-content type as a=accept-wrapped-types:*;

vi. The CPM Client SHALL set the a=setup attribute as "active";

vii. SHALL include an SDP 'msrp-cema' attribute in the MSRP media description of the SDP.

NOTE: The value '*' for 'accept-wrapped-types ' attribute is for example and implementations may copy the value from the SDP offer or don't have to include this attribute in the SDP answer.

d. The CPM Client SHALL include any of the Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID header fields received in the SIP INVITE request;

e. SHALL include a Session-Expires header field in the SIP 200 "OK" response to the SIP INVITE request with the refresher parameter set based on the value received in the SIP INVITE request and SHALL start the SIP session timer according to the rules and procedures specified in [RFC4028];

f. SHALL include the option tag 'timer' in the Supported header field and in the Required header field;

3. SHALL send the SIP 200 "OK" response according to the rules and procedures of the SIP/IP core.

Upon receiving a SIP CANCEL request, the CPM Client SHALL act as UAS to handle the request according to the rules and procedures of [RFC3261].

Upon receiving a SIP ACK request, the CPM Client:

1. SHALL handle the SIP ACK request according to the rules and procedures of [RFC3261];
2. SHALL act as an MSRP client according to [RFC6135];
3. SHALL act as an active endpoint to open the transport connection according to [RFC6135];
4. SHALL establish the MSRP connection according to the MSRP connection parameters in the SDP offer (c/m-line address information) received in the SIP INVITE request according to [RFC6714];
5. SHALL send an empty MSRP SEND request to bind the MSRP connection to the MSRP session from the perspective of the passive endpoint according to the rules and procedures of [RFC4975] and [RFC6135].

When the CPM Client receives an MSRP request, the Client SHALL follow the rules and procedures defined in [RFC4975] and in [RFC6714]. If an MSRP SEND request indicates the use of chunking, the CPM Client SHALL wait until all further MSRP SEND requests for the remaining chunks have been received and SHALL reassemble the entire set of MSRP requests into the CPM Standalone Message.

The CPM Client SHALL handle the CPIM message of the received CPM Standalone Message as described in section 7.2.2.3 "*CPIM Handling of Received CPM Standalone Messages and SIP IMDNs*".

If the session is terminated before the complete CPM Standalone Message was received or the TCP connection used for the MSRP session is lost before the transmission was completed, the received contents SHALL be silently discarded.

Upon receiving a SIP BYE request, the CPM Client SHALL respond to the SIP BYE request as described in [3GPP TS24.229] and SHALL release the media resources. If the CPM Message was stored in the CPM Message Store by the CPM Participating Function on the terminating side, the CPM Client SHALL retrieve from the SIP BYE request the Message-UID header field containing the UID value of the CPM Message stored in the CPM Message Store. This value is used further by the CPM Client in the message synchronization process with CPM Message Store via IMAP, for correlation with the CPM Messages handled via the SIP channel.

If a Message-UID header field is not present in the SIP BYE request, or the SIP BYE request fails to arrive at the CPM Client, the CPM Client SHALL identify when message duplicates exist locally in the client (e.g. received via the SIP delivery and also received via IMAP synchronization) based on:

a. the IMAP folder in which the message is stored (i.e. folder SHALL match the identity of the other party in a 1-1 CPM Standalone Message and the Conversation-ID for the CPM Group Standalone Message) and,

b. the CPM SIP headers fields of a message delivered via SIP match the values of the MIME headers of message objects received during the IMAP synchronization, as described in the table 2 section 7.2.1.1 "*Sending a Pager Mode CPM Standalone Message*".

7.2.2.3 CPIM Handling of Received CPM Standalone Messages and SIP IMDNs

When a CPM Client has received a CPM Standalone Message or a SIP IMDN (via SIP MESSAGE request), the CPM Client SHALL check the CPIM body of the SIP MESSAGE request, or of the MSRP SEND of a Large Message Mode CPM Standalone Message:

1. SHALL check the Content-Disposition header field of the CPIM message. If there is a value in the header field, the terminating CPM Client SHALL follow the value;
2. If the body includes references to content external to the message having the Content-Type: message/external-body; the CPM Client SHALL treat them according to rules and procedures in [RFC4483];
3. If the service provider policy is to support direct delivery procedure for the originating CPM Message, the CPM Client SHALL check the CPIM Message-Direction header field presence, as follows:
    a. If CPIM Message-Direction header field is not present, continue with step 4;
    b. If CPIM Message-Direction header field is present and set to the value “sent” in the CPIM wrapper of the payload. The CPM client SHALL check the following:
        i. In case of CPM Standalone Message is generated by the same CPM User from another CPM Client; the Request-URI is set as described in section 6.1.2 “*Identifying the recipient device in SIP requests and responses”*;
        ii. The CPM client should ignore the IMDN headers in CPIM in this case;
        iii. In case IMDN is “displayed” from another CPM Client of the CPM User, the receiving CPM Client SHALL use this as indication that the message was “Seen” by the CPM User on another CPM Client;
4. If the Content-Type inside the CPIM message is not set to “message/imdn+xml” as specified in [RFC5438], then the CPM Client SHALL check the IMDN headers to determine whether the incoming CPM Standalone Message contains the request for disposition notifications. If true, the CPM Client SHALL send a disposition notification as described in:
    a. section 5.4.1 “*Generate Delivery Notification*”, if a delivery notification was requested;
    b. section 2 “*Generate Read Report*”, if a display notification was requested and the message was displayed to the CPM User;
5. If the Content-Type of the CPIM message is set to “message/imdn+xml” as specified in [RFC5438], then the CPM Client SHALL handle the received IMDN notification based on the type of the notification received in the body:
    a. If the IMDN notification contains a delivery notification, the CPM Client SHALL handle it as described in section 5.4.3 “*Receive Delivery Notification*”;
    b. If the IMDN notification contains an interworking notification, the CPM Client SHALL handle it as described in section 5.4.7 “*Receive Interworking Notification*”;
    c. If the IMDN notification contains a display notification, the CPM Client SHALL handle it as described in section 5.4.4 “*Receive Read Report*”.
6. SHALL check if the CPIM cpm.Payload-Type header field is present, and if so then it SHALL determine if it supports all the Content-Types values listed for the included message body. If the CPM Client:
    a. identifies one or more content type that it does not understand, then it SHALL ignore those;
    b. otherwise, if it determines that it does not support any of the listed content types, it MAY reject the message depending on the case: with a SIP 488 “Not acceptable here” if it received a SIP MESSAGE request, or with a MSRP 415 “Media not understood” if it received a MSRP SEND request.

7.2.3 Deferred CPM Message Handling

7.2.3.1 Subscribe to Deferred CPM Message Info

If device settings are set to pull Deferred CPM Standalone Messages, the CPM Client SHALL subscribe to the “deferred-messages” event package in order to receive information about Deferred CPM Standalone Messages. The CPM Client:

1. SHALL generate a SIP SUBSCRIBE request, according to the rules and procedures of [RFC6665] with the following clarifications:

a. SHALL set the Request-URI to the address of the CPMDeferredMsgMgmtURI configured for the CPM service in the home CPM network of the CPM User (e.g., CPMDeferredMsgMgmt@hostname) as provisioned in the CPM Client;

b. SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred' percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI"*;

c. SHALL include a Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred' 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 of the Contact header field;

d. MAY include one of the registered CPM Addresses of the CPM User as described in section 6.1 "*Authenticated Originator's CPM Address*" as the authenticated originator's CPM Address;

e. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred';

f. SHALL set the Event header field to "deferred-messages";

g. SHALL set the Expires header field to "0";

h. SHALL send the SUBSCRIBE request according to the rules and procedures of the SIP/IP core.

Upon receiving a SIP NOTIFY request with the information on the Deferred CPM Standalone Messages, the CPM Client either:

- SHALL display the list of Deferred CPM Standalone Messages to the CPM User and SHALL allow the CPM User to process the list as described in section 7.2.3.2.1 "*Handling Deferred CPM Message(s) before Having Received an Expiry Notification*" to execute the decisions of the CPM User; or
- SHALL fetch all Deferred CPM Standalone Messages as described in section 7.2.3.2.1 "*Handling Deferred CPM Message(s) before Having Received an Expiry Notification*" if that is allowed by its local settings.

7.2.3.2 Handling Deferred CPM Message(s)

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

Handling of Deferred CPM Standalone Message(s) is performed by setting up a session with the CPM Participating Function. The message-URI-ID(s) of the Deferred CPM Standalone Message(s) to be retrieved, deleted, interworked or stored are listed in a URI list as defined in [RFC5366]. The absence of a URI-list in the session set up indicates that all messages are to be retrieved.

NOTE 1: The CPM Client may initiate this procedure at the instruction of the CPM User or based on local settings and circumstances.

When handling Deferred CPM Standalone Message(s) before receiving a notification about expiry, the CPM Client:

1. SHALL generate an initial SIP INVITE request according to the rules and procedures of [RFC3261];
2. SHALL set the Request-URI of the SIP INVITE request to CPMDeferredMsgMgmtURI for the CPM service in the home CPM network of the CPM User;
3. SHALL include the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred' 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 of the Contact header field;
4. SHALL include the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred' 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 of the Accept-Contact header field;
5. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred';
6. SHALL include in the SIP INVITE request a MIME SDP body as an SDP offer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975] and [RFC6714] with the following additional clarifications. The CPM Client:

   a. SHALL set the SDP "accept-types" attribute to a=accept-types:message/cpim;

   b. MAY list other formats or use '*' as defined in [RFC4975];

   c. SHALL set the media direction attribute to a=recvonly;
   d. SHALL include an SDP 'msrp-cema' attribute in the MSRP media description of the SDP.

7. If not all the Deferred CPM Standalone Messages are to be retrieved, the CPM Client
   a. SHALL include a MIME resource-list body as specified in [RFC5366]. Each entry in the URI-list identifies a Deferred CPM Standalone Message and SHALL contain one of the following parameters:
      i. “?cpm_action=deliver”, if the Deferred CPM Standalone Message identified by the URI is to be retrieved;
      ii. “?cpm_action=delete”, if the Deferred CPM Standalone Message identified by the URI is to be deleted;
      iii. “?cpm_action=interwork”, if the Deferred CPM Standalone Message identified by the URI is to be interworked;
      iv. “?cpm_action=store”, if the Deferred CPM Standalone Message identified by the URI is to be delivered to the Message Storage Server.

   NOTE 2: Any deferred message not mentioned in the MIME resource-list in the SIP INVITE request will stay deferred.

8. SHALL send the SIP INVITE request to the CPM Participating Function according to the rules and procedures of [RFC3261] and the SIP/IP core.

   NOTE 3: The media plane of the above session is not needed and therefore is not being set up.

On receiving a SIP 488 “Not Acceptable Here” response with a SIP Warning header field with the warning text set to “125 No messages” the CPM Client SHALL not pursue handling of the Deferred CPM Standalone Messages specified in above SIP INVITE request and SHALL process the SIP 488 “Not Acceptable Here” response according to the rules and procedures of [RFC3261].

Any other error response SHALL be processed according to the rules and procedures of [RFC3261].

On receiving a SIP 200 “OK” response to the SIP INVITE request, the CPM Client SHALL generate and send a SIP ACK request according to [RFC3261]. After having sent the SIP ACK request, the CPM Client SHALL send a SIP BYE request.

If there are any Deferred CPM Standalone Messages to be delivered, these will be sent by the CPM Participating Function and the CPM Client SHALL handle the received CPM Standalone Message as described in section 7.2.2 “*Receiving CPM Standalone Messages*”.

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

When retrieving CPM Standalone Message(s) after receiving a notification that Deferred CPM Standalone Message(s) were expired as in section 7.2.3.3 “*Receiving a notification for Expiry of Deferred CPM Message”*, the CPM Client SHALL perform the following procedures:

1. If the Deferred CPM Standalone Messages were discarded after expiry, the CPM Client SHALL inform the CPM User that the Deferred CPM Standalone Messages were already discarded because they were expired;
2. If the Deferred CPM Standalone Messages were stored in Message Storage Server after expiry, the CPM Client SHALL perform the following procedures:
   a. MAY send a request to fetch the stored messages as described in section 6.3.2 “*Object Fetch Operation*” of [OMA-CPM_TS_MessageStorage]. The request SHALL use the UID which was obtained in section 7.2.3.3 “*Receiving a notification for Expiry of Deferred CPM Standalone Message”*; or, alternatively
   b. MAY inform the CPM User that the Deferred CPM Standalone Messages were already stored in Message Storage Server because they were expired.

7.2.3.3 Receiving a notification for Expiry of Deferred CPM Standalone Message

When receiving a notification that includes information on the handling of deferred messages that expired (i.e. whether those messages discarded or stored), the CPM Client:

1. MAY inform the CPM User that the messages expired and were processed according to the user preferences;
2. SHALL store the UID(s) included in the notification, if any.

   NOTE: The CPM Client can later fetch the corresponding messages from the Message Storage Server.

7.2.3.4 Notification Handling

7.2.3.4.1 Out-of-band Notifications

NOTE: The out-of-band notifications reach the CPM Client, when it is not registered in SIP/IP Core for the CPM service, via the Push Client of the device.

Upon receiving an out-of-band notification from the CPM Participating Function to signal that there is a new unseen Deferred CPM Standalone Message available for the CPM Client, the CPM Client:

1. MAY display the received notification to the CPM User to verify whether the CPM User wants to connect to the CPM service;

   NOTE: The CPM Client may also, e.g. based on device settings, take a decision on behalf of the CPM User, without interacting with the CPM User.
2. SHALL decide, possibly based on the CPM User's input, whether to connect to the CPM service. If yes, then the CPM Client SHALL register to the CPM service as described in section 7.1 "*Registering at the SIP/IP Core*".

7.2.3.4.2 In-band Notifications

Upon receiving a SIP MESSAGE request containing the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg' included in the Accept-Contact header field for in-band notification to signal that there are Deferred CPM Standalone Messages available for the CPM Client, the CPM Client MAY, depending on device settings, display the received notification to the CPM User to allow him to process the notification as described in section 7.2.3.2.1 "*Handling Deferred CPM Standalone Message(s) before Having Received an Expiry Notification*".

7.3 CPM Session Handling

7.3.1 Initiating New CPM Sessions

7.3.1.1 Initiating a CPM 1-1 Session

The CPM Client SHALL generate an initial SIP INVITE request according to the rules and procedures of [RFC3261]. In this SIP INVITE request, the CPM Client:

1. SHALL include the address of a target CPM User or a non-CPM Principal in the Request-URI;
2. SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' , 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;;
3. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session';
4. SHALL include a Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' , 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;
5. SHALL include the CPM Address of the CPM User as the preferred originator's CPM Address, as described in section 6.1 "*Authenticated Originator's CPM Address*";
6. SHALL include a User-Agent header field to indicate the OMA CPM release version of the CPM Client as specified in Appendix D "*Release Version in User-agent and Server headers*";
7. If anonymity is requested, SHALL include value "id" in a Privacy header field according to the rules and procedures of [RFC3325];
8. SHALL include a Supported header field with the option tag 'timer'; and,
9. MAY include in the Supported header field the option tag 'gruu';
10. SHALL NOT include a Session-Expires header field according to the rules and procedures of [RFC4028];
11. SHALL NOT include the Min-SE header field according to the rules and procedures of [RFC4028];
12. If the CPM Session starts a new CPM Conversation, the CPM Client SHALL include a Conversation-ID header field set to a newly generated value as specified in Appendix C "*CPM-defined Header fields*"; Otherwise include the Conversation-ID of the existing CPM Conversation;

13. The CPM Client SHALL include a Contribution-ID header field as specified in Appendix C "*CPM-defined Header fields*" with:
    a. the Contribution-ID value from a previous 1-1 CPM Session that was established and ended with this recipient, if one exists; or
    b. a newly generated Contribution-ID value otherwise;
14. If the CPM Session is in reply to a previously received CPM Standalone Message, CPM File Transfer or CPM Session Invitation, the CPM Client SHALL include the InReplyTo-Contribution-ID header field with the value of Contribution-ID in the previously received CPM Standalone Message, CPM File Transfer or CPM Session Invitation;
15. SHALL include a MIME SDP body as an SDP offer as described in section 5.2.1.1 "*SDP Contents when Initiating or Modifying a CPM Session*";
16. SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

On receipt of a SIP 200 "OK" response to the initial SIP INVITE request the CPM Client SHALL handle the response according to the rules and procedures of [RFC3261], with the following clarifications:

1. The CPM Client SHALL start a SIP session timer using the value received in the Session-Expires header field according to the rules and procedures of [RFC4028].
2. The CPM Client SHALL generate and send a SIP ACK request as an acknowledgement of the final response according to the rules and procedures of [RFC3261].
3. The CPM Client SHALL initiate the Media Plane as described in section 7.3.9 "*Media Plane Handling for CPM Sessions*".

On receiving other non-2XX final responses to the sent SIP INVITE request, the CPM Client:

A. SHALL generate and send a SIP ACK request as an acknowledgement of the final response, according to the rules and procedures of [RFC3261];
B. MAY inform the CPM User that the session could not be set up.

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

The CPM Client SHALL check that the number of invited CPM Users on the URI-list does not exceed the maximum number of Participants allowed in a CPM Group Session for a CPM Ad-hoc Group, as configured to the device. If exceeded, the CPM Client SHALL notify the CPM User and SHALL NOT continue with the procedure below.

The CPM Client SHALL generate an initial SIP INVITE request according to the rules and procedures of [RFC3261]. In this SIP INVITE request the CPM Client:

1. SHALL set the Request-URI of the SIP INVITE request to the CPM Controlling Function URI provisioned in the device;
2. SHALL include an Accept-Contact header field with the CPM feature-tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session', 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;
3. SHALL include a Contact header field with the CPM feature-tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session', 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;
4. SHALL include the CPM Address of the CPM User as the preferred originator's CPM Address, as described in section 6.1 "*Authenticated Originator's CPM Address*";
5. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group';
6. SHALL include a MIME resource-list body with the invited CPM Users as specified in [RFC5366] from which only the mandatory parts are applicable and where the "CC" and "BCC" copy control identifiers SHALL NOT be used;
7. If anonymity is requested by the CPM User, SHALL include value "id" in the Privacy header field according to the rules and procedures of [RFC3325];
8. MAY include a Subject header field containing the text selected for this CPM Group Session;

9. If the CPM Session starts a new CPM Conversation, the CPM Client SHALL include a Conversation-ID header field set to a newly generated value as specified in Appendix C "*CPM-defined Header fields*"; Otherwise include the Conversation-ID of the existing CPM Conversation;
10. The CPM Client SHALL include a Contribution-ID header field with a newly generated Contribution-ID value as specified in Appendix C "*CPM-defined Header fields*";
11. If the CPM Session is in reply to a previously received CPM Standalone Message, CPM File Transfer or CPM Session Invitation, the CPM Client SHALL include the InReplyTo-Contribution-ID header field with the value of Contribution-ID in the previously received CPM Standalone Message, CPM File Transfer or CPM Session Invitation;
12. SHALL include a MIME SDP body as a SDP offer as described in section 5.2.1.1 "*SDP Contents when Initiating or Modifying a CPM Session*";
13. If the CPM User requested the CPM Group Session to be a CPM Closed Group Session, and the CPM Client supports CPM Closed Group Session, then the CPM Client SHALL populate the SDP attribute *a=chatroom* defined in [RFC7701] with the CPM reserved chat-token value of '*org.openmobilealliance.groupchat.closed*' to indicate the request for a CPM Closed Group Session in the SIP INVITE. The SDP attribute value SHALL be: "*a=chatroom:org.openmobilealliance.groupchat.closed"*;
14. If the CPM Group Session Data management is supported, it SHALL include the a=accept-wrapped-types attribute set to the value of the CPM Group Session Data Content-Type, as defined in section 6.8 "*CPM Group Session Data Management*" and the value "multipart/related";
15. If Participant Information Notifications via Conference Information Object is supported, it SHALL include in the a=accept-wrapped-types attribute the value application/conference-info+xml, and the value "multipart/related" (if not already present);
16. SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

Upon receiving a SIP final response to the SIP INVITE request, the CPM Client SHALL handle the response according to the rules and procedures of [RFC3261], with the following clarifications:

1. The CPM Client SHALL generate and send a SIP ACK request as an acknowledgement of the final response according to the rules and procedures of [RFC3261];
2. If a SIP 200 "OK" response was received:
   a) the CPM Client SHALL store the contents of the Contact header field as the CPM Group Session Identity, as described in [RFC4579];
   b) If it populated the SDP attribute *a=chatroom* defined in [RFC7701] with the CPM reserved closed chat-token value of '*org.openmobilealliance.groupchat.closed*' to indicate support for a CPM Closed Group Session in the SIP INVITE, then it SHALL check if the same attribute value was received in the SIP response;
      i. If the SDP attribute is not present in 200 "OK", or if it is present but does not have the value of '*org.openmobilealliance.groupchat.closed*' to indicate support for the CPM Closed Group Session, then the CPM Client SHALL handle the CPM Group Session as an open CPM Group Session (e.g., providing the CPM User with the option to voluntary leave the open CPM Group Session, or to add and remove Participants if both the CPM Client and the CPM Controlling Function support the CPM Group Session Data management);
      ii. Otherwise, if the SDP attribute is present in 200 "OK" with the same value of '*org.openmobilealliance.groupchat.closed*', the CPM Client SHALL handle the CPM Group Session as closed;
   c) If the CPM Client did not populate the SDP attribute *a=chatroom* defined in [RFC7701] with the CPM reserved closed chat-token value of '*org.openmobilealliance.groupchat.closed*', but it has received it in the SIP 200 "OK" response indicating that the CPM Group Session is Closed, then if it supports CPM Closed Group Session it SHALL handle it as closed. Otherwise, if the CPM Client does not support CPM Closed Group Session it SHALL handle the CPM Group Session as open.
   d) The CPM Client SHALL initiate the Media Plane as in section 7.3.9 "Media Plane Handling for CPM Sessions", with the following addition:
      i. If the SDP answer included the a=accept-wrapped-types attribute, set to the value of the CPM Group Session Data Content-Type as defined in section 6.8 "*CPM Group Session Data*

Management", and the CPM Client supports also the CPM Group Session Data management , then the CPM Client SHALL consider that the CPM Group Session Data management is supported in this CPM Group Session;

ii. If the SDP answer included in the a=accept-wrapped-types attribute the value application/conference-info+xml and if the CPM Client supports Participant Information Notifications via Conference Information Object, then the CPM Client SHALL consider that Participant Information Notifications via Conference Information Object is applicable in this CPM Group Session;

e) If Participant Information Notifications via Conference Information is not applicable in this CPM Group Session, then the CPM Client SHALL subscribe to the conference state event package of the CPM Group Session Identity as specified in section 7.3.10 "*Participant Information*" and SHALL maintain the subscription throughout the CPM Client's participation in the CPM Group Session.

3. If other non-2XX final responses to the sent SIP INVITE request are received the CPM Client MAY inform the CPM User that the session could not be set up.

7.3.1.3 Initiating a CPM Group Session for a CPM Pre-defined Group

The CPM Client SHALL generate an initial SIP INVITE request according to the rules and procedures of [RFC3261]. In this SIP INVITE request the CPM Client:

1. SHALL set the Request-URI of the SIP INVITE request to the CPM Pre-Defined Group address;
2. SHALL include an Accept-Contact header field with the CPM feature-tag 'urn:urn-7:3gpp-service.ims.'urn:urn-7:3gpp-service.ims.icsi.cpm.session', 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;
3. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group';
4. SHALL include a Contact header field with the CPM feature-tag 'urn:urn-7:3gpp-service.ims.'urn:urn-7:3gpp-service.ims.icsi.cpm.session', 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;
5. SHALL include the CPM Address of the CPM User as the preferred originator's CPM Address, as described in section 6.1 "*Authenticated Originator's CPM Address*";
6. If anonymity is requested by the CPM User, the CPM Client SHALL include value "id" in the Privacy header field according to the rules and procedures of [RFC3325];
7. If the CPM Session starts a new CPM Conversation, the CPM Client SHALL include a Conversation-ID header field set to a newly generated value as specified in Appendix C "*CPM-defined SIP Headers*"; Otherwise include the Conversation-ID of the existing CPM Conversation;
8. The CPM Client SHALL include a Contribution-ID header field with a newly generated Contribution-ID value as specified in Appendix C "*CPM-defined SIP Headers*";
9. If the CPM Session is in reply to a previously received CPM Standalone Message, CPM File Transfer or CPM Session Invitation, the CPM Client SHALL include the InReplyTo-Contribution-ID header field with the value of Contribution-ID in the previously received CPM Standalone Message , CPM File Transfer or CPM Session Invitation;
10. SHALL include in the SIP INVITE request a MIME SDP body as a SDP offer as described in section 5.2.1.1 "*SDP Contents when Initiating or Modifying a CPM Session*";
11. SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

On receiving a SIP 200 "OK" response to the SIP INVITE request, the CPM Client SHALL handle the response according to the rules and procedures of [RFC3261], with the following clarifications:

1. The CPM Client SHALL generate and send a SIP ACK request as an acknowledgement of the final response towards the CPM Controlling Function; and,
2. The CPM Client SHALL store the contents of the Contact header field as the CPM Group Session Identity, as described in [RFC4579]; and,
3. The CPM Client SHALL initiate the Media Plane as in section 7.3.9 "*Media Plane Handling for CPM Sessions*" with the following additions:

i. If the SDP answer included in the a=accept-wrapped-types attribute the value “application/conference-info+xml” and if the CPM Client supports Participant Information Notifications via Conference Information Object, then the CPM Client SHALL consider that Participant Information Notifications via Conference Information Object is applicable in this CPM Group Session;

If Participant Information Notifications via Conference Information is not applicable in this CPM Group Session, then the CPM Client SHALL subscribe to the conference state event package of the CPM Group Session Identity as specified in section 7.3.10 “*Participant Information*” and SHALL maintain the subscription throughout the CPM Client’s participation in the CPM Group Session. In case of failure to set a subscription (error response received to the SIP SUBSCRIBE request), the CPM Client MAY carry on the CPM Group Session but it might not be able to provide support for the CPM Long-lived Group Session, hence the CPM User may not be able to re-start it if the last Participant Information is not delivered to the client by other means (e.g., if CPM Participating Function will not deliver it).

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

The CPM Client SHALL generate a SIP INVITE request according to the rules and procedures of [RFC3261] . In this SIP INVITE request the CPM Client:

1. SHALL set the Request-URI to the address of the target CPM Pre-defined Group;
2. SHALL include an Accept-Contact header field with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ 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;
3. SHALL include a Contact header field with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ 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;
4. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group’;
5. SHALL include the CPM Address of the CPM User as the preferred originator's CPM Address, as described in section 6.1 “*Authenticated Originator’s CPM Address*”;
6. SHALL include a User-Agent header field to indicate the OMA CPM release version of the CPM Client as specified in Appendix D “*Release Version in User-agent and Server headers*”
7. If anonymity is requested, SHALL include value "id" in a Privacy header field according to the rules and procedures of [RFC3325].

   NOTE: If anonymity is not allowed for the CPM Group indicated with the Request-URI of the SIP INVITE based on the rules specified in the [OMA-XDM-Group] the CPM Session Invitation will not be allowed by the CPM Controlling Function hosting the CPM Group.
8. SHALL include a Conversation-ID header field set to a newly generated value as specified in Appendix C “*CPM-defined Header fields*”;
9. SHALL include a Contribution-ID header field with a newly generated Contribution-ID value as specified in Appendix C “*CPM-defined SIP Headers*”;
10. SHALL include a MIME SDP body as a SDP offer as described in section 5.2.1.1 “*SDP Contents when Initiating or Modifying a CPM Session*”, with the following addition:

    a. If the CPM Group Session Data management is supported, it SHALL include the a=accept-wrapped types attribute set to the value of the CPM Group Session Data Content-Type as specified in section 6.8 “*CPM Group Session Data Management*” and the value “multipart/related”;

    b. If Participant Information Notifications via Conference Information Object is supported, it SHALL include in the a=accept-wrapped-types attribute the value “application/conference-info+xml”, and the value “multipart/related” (if not already present);
11. SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

On receiving a SIP 200 "OK" response to the SIP INVITE request the CPM Client SHALL handle the response according to the rules and procedures of [RFC3261], with the following clarifications:

a) The CPM Client SHALL generate and send a SIP ACK request as an acknowledgement of the final response towards the CPM Controlling Function;

b) The CPM Client SHALL store the contents of the Contact header field as the CPM Group Session Identity, as described in [RFC4579];

c) The CPM Client SHALL store the CPM Conversation Identity and CPM Contribution Identity, if received, and use them for CPM Conversation identification instead of the newly generated CPM Conversation Identity and CPM Contribution Identity included in the SIP INVITE request;

d) The CPM Client SHALL initiate the Media Plane as in section 7.3.9 "*Media Plane Handling for CPM Sessions*".

e) If the SDP answer included the a=accept-wrapped-types attribute set to the value of the CPM Group Session Data Content-Type, as specified in section 6.8 "*CPM Group Session Data Management*" and if the CPM Client supports also the CPM Group Session Data management , the CPM Client SHALL consider that the CPM Group Session Data management is supported in this CPM Group Session;

f) If the SDP answer included in the a=accept-wrapped-types attribute the value application/conference-info+xml and if the CPM Client supports Participant Information Notifications via Conference Information Object, then the CPM Client SHALL consider that Participant Information Notifications via Conference Information Object is applicable in this CPM Group Session;

If Participant Information Notifications via Conference Information Object is not applicable in this CPM Group Session, then the CPM Client SHALL subscribe to the conference state event package of the CPM Group Session Identity as specified in section 7.3.10 "*Participant Information*" and SHALL maintain the subscription throughout the CPM Client's participation in the CPM Group Session.

7.3.1.5 Re-joining a CPM Long-lived Group Session

If the CPM Client did not receive the CPM Group Session information via the delivery from CPM Participating Function (e.g. due to connectivity) and therefore has retrieved the CPM Group Session information from the CPM Message Store, then:

A) If a Group State Object and/or Conference Information Object is available (i.e. a CPM Group Session Identity is available):

  a. It SHALL check whether the CPM User's identity is still among the participants of the CPM Group Session. If the identity of the CPM User is not listed among the participants, then the CPM Client SHALL NOT continue with the Re-join procedures;

  b. it SHALL use the information from the Session Info Object to populate the SIP header fields of the SIP INVITE request to join in the CPM Group Session; and,

  c. the last Group State Object information and/or Conference Information Object (based on the timestamp) to retrieve the CPM Group Session Identity (focus URI) and to show to the CPM User the last active Participants in the CPM Group Session;

  d. if Participant Information Notification via Conference Information Object is not applicable, SHALL send a SIP SUBSCRIBE to the conference event for that CPM Group Session Identity, and once it received the SIP NOTIFY containing the latest Participants list, subject and icon, it SHALL use that list to inform the CPM User of the latest list of Participants, subject, icon and bulletin of the CPM Group Session, instead of using the information from Group State Object;

B) If a Group State Object or Conference Information Object is not available, then the CPM Client MAY start a new CPM Group Session as described in section 7.3.1.2 "*Initiating a CPM Group Session for a CPM Ad-hoc Group*" with the following clarifications:

  a. SHALL compose the Participant list as follows:

    i. SHALL add the CPM users extracted from the <invited-participants> element in the body of the Session Info Object; and

    ii. SHALL use the From MIME header of the Session Info Object to extract the originator of that CPM Group Session and add him/her to the Participant list;

    iii. SHALL remove his/her own address from the Participant list.

Regardless of whether the CPM Client received the Group Session information via delivery from the CPM Participating Function or it has retrieved it from the CPM Message Store, the CPM Client SHALL generate a SIP INVITE request according to the rules and procedures of [RFC3261]. In this SIP INVITE request the CPM Client:

1) SHALL set the Request-URI to the address of the target CPM Group Session of CPM Long-live Group Session;

2) SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' 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;

3) SHALL include a Contact header field with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ 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;;
4) SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group’;
5) SHALL include the CPM Address of the CPM User as the preferred originator's CPM Address in Contact header field, as described in section 6.1 “*Authenticated Originator’s CPM Address*”;
6) SHALL include a User-Agent header field to indicate the OMA CPM release version of the CPM Client as specified in Appendix D “*Release Version in User-agent and Server headers*”
7) If anonymity is requested, SHALL include value "id" in a Privacy header field according to the rules and procedures of [RFC3325].

NOTE: If anonymity is not allowed for the CPM Group indicated with the Request-URI of the SIP INVITE based on the rules specified in the [OMA-XDM-Group] the CPM Session Invitation will not be allowed by the CPM Controlling Function hosting the CPM Group.

8) SHALL include the Conversation-ID header field of the requested CPM Group Session;
9) SHALL include the Contribution-ID header field of the requested CPM Group Session;
10) SHALL include the MIME SDP body of the requested CPM Group Session as a SDP offer as described in section 5.2.1.1 “*SDP Contents when Initiating or Modifying a CPM Session*”, with the following addition:
    a) If the CPM Group Session Data management is supported, it SHALL include in the a=accept-wrapped-types attribute the value of the CPM Group Session Data Content-Type as specified in section 6.8 “*CPM Group Session Data Management*” and the value “multipart/related”;
    b) If Participant Information Notifications via Conference Information Object is supported, it SHALL include in the a=accept-wrapped-types attribute the value “application/conference-info+xml” and the value “multipart/related” (if not already present);
11) SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

On receiving a SIP 200 "OK" response to the SIP INVITE request the CPM Client SHALL handle the response according to the rules and procedures of [RFC3261], with the following clarifications:

a) The CPM Client SHALL generate and send a SIP ACK request as an acknowledgement of the final response towards the CPM Controlling Function;
b) The CPM Client SHALL store the contents of the Contact header field as the CPM Group Session Identity, as described in [RFC4579];
c) The CPM Client SHALL initiate the Media Plane as in section 7.3.9 “*Media Plane Handling for CPM Sessions*”, with the following clarification:
    i) If the SDP answer included the a=accept-wrapped-types attribute set to the value of the CPM Group Session Data Content-Type as specified in section 6.8 “*CPM Group Session Data Management*” and the CPM Client supports also the CPM Group Session Data management , the CPM Client SHALL consider that the CPM Group Session Data management is supported in this CPM Group Session;
    ii) If the SDP answer included in the a=accept-wrapped-types attribute the value application/conference-info+xml and if the CPM Client supports Participant Information Notifications via Conference Information Object, then the CPM Client SHALL consider that Participant Information Notifications via Conference Information Object is applicable in this CPM Group Session;

On receiving SIP 404 “Not Found” or SIP 403 ”Forbidden“ error response to the SIP INVITE request, the CPM Client SHALL generate and send a SIP ACK request as an acknowledgement of the final response towards the CPM Controlling Function. If the CPM Client has to continue the CPM Group Session for the CPM User, then the CPM Client MAY generate and send a new SIP INVITE request as specified in section 7.3.1.2 “*Initiating a CPM Group Session for a CPM Ad-hoc Group*”, with the following clarifications:

i. SHALL include a MIME resource-list body with the last Participant Information as specified in [RFC5366];

ii. SHALL include the Conversation-ID header field of the requested CPM Group Session;

iii. SHALL include the Contribution-ID header of the requested CPM Group Session.

In the case of CPM Client receives a SIP INVITE of the same CPM Group Session during it own SIP INVITE of re-joint CPM Group Session, CPM Client SHALL use the received SIP INVITE request as specified in section 7.3.2 "*Receiving a CPM Session Invitation*", instead of it own initiated SIP INVITE request.

If the CPM Group Session was successfully established with the CPM Client and if Participant Information Notifications via Conference Information Object is not applicable in this CPM Group Session, then the CPM Client SHALL subscribe to the conference state event package of the CPM Group Session Identity as specified in section 7.3.10 "*Participant Information*" and SHALL maintain the subscription throughout the CPM Client's participation in the CPM Long-lived Group Session.

7.3.2 Receiving a CPM Session Invitation

Upon receiving an initial SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' included in the Accept-Contact header field, the CPM Client:

1. SHALL return:
   a. a SIP 480 "Temporarily Unavailable" response if there are not enough resources to handle the CPM Session, or if the CPM User or CPM Client temporarily declines the CPM Group Session invitation (e.g. CPM User is roaming and does not want to take the CPM Group Session at that time) or
   b. a SIP 486 "Busy Here" response if the CPM User or CPM Client decides not to take the additional session (e.g. if audio session is requested when audio is already part of another session) or
   c. a SIP 302 "Moved Temporarily" response if redirecting the CPM Session Invitation is requested by a local device's setting or CPM User, or
   d. a SIP 603 "Declined" response if the CPM User has rejected the invitation.

   Otherwise, continue with the rest of steps;
2. If a Session-Replaces header (as defined in Appendix C "*CPM-defined SIP Headers*") is included in the SIP INVITE request, it SHALL determine whether a corresponding CPM 1-1 Session exists. If there is no ongoing CPM 1-1 Session whose Contribution-ID is identical to the value indicated in the Session-Replaces header, the CPM Client SHALL return a SIP 481 "Call/Transaction Does Not Exist" response.

   Otherwise, continue with rest of steps:
3. It SHALL check that the SIP INVITE SDP directional media attribute is set to "a= sendrecv", in which case the CPM Client SHALL use this CPM Session:
   a. to receive CPM Chat Messages,
   b. to send back any requested disposition notifications via MSRP (i.e. delivery notifications, and if applicable display notifications) as described in sect. 5.4.1 "*Generate Delivery Notifications*" and sect. 2 "*Generate Read Report*"; and
   c. to send CPM Chat Messages and to receive their associated disposition notifications, as described in sect. 5.4.3 "*Receive Delivery Notifications*" and sect. 5.4.4 "*Receive Read Report*";
4. Else, if the SDP directional media attribute is not set to "a= sendrecv", the CPM Client SHALL return SIP 488 "Not Acceptable Here" response according to the rules and procedures of [RFC3261] and SHALL not continue with the remaining steps;
5. if the received SIP INVITE request includes a Referred-By header field, the CPM Client SHALL determine that the CPM Group Session invitation was generated following the CPM User addition by another Participant, as described in section 7.3.5 "*Invite other Principals to existing CPM Group Session*". The CPM Client MAY use the Referred-By header field value for rendering purposes to the CPM User, to identify the inviting Participant;
6. MAY render the CPM Session Invitation to the CPM User based on a local device settings, with the following clarifications:
   a. The CPM Client MAY render the subject header field as part of the CPM Session Invitation, if present;
   b. If a Session-Replaces header field is received, the CPM Client MAY render an indication that the CPM Group Session is a replacement for a CPM 1-1 Session;
   c. The CPM Client MAY display to the invited CPM User the CPM Address of the inviting CPM User, if privacy rules allow it. The CPM Client SHALL NOT display the CPM address of the inviting CPM User if Privacy header field includes value 'id';

   d. MAY display to the CPM User a list of the other invited users if the SIP INVITE carried a recipient-list-history body;
   e. The CPM Client SHALL return a SIP 180 “Ringing” provisional response;

   Otherwise, continue with the rest of the steps;

7. Otherwise, if the CPM Session request is to be accepted (e.g. subject to CPM User acceptance, or to auto-accept type of configurations on the CPM Client), it then the CPM Client:
   a. SHALL return a SIP 180 “Ringing” provisional response if auto-accept does not apply;
   b. SHALL store the contents of the Contact header field as the CPM Group Session Identity, as described in [RFC4579];
   c. If a Session-Replaces header field is received, it SHALL replace the existing 1-1 CPM Session with the new CPM Group Session;
   d. If the SDP offer included the a=accept-wrapped-types attribute set to the value of the CPM Group Session Data Content-Type, as specified in section 6.8 “*CPM Group Session Data Management*” the CPM Client SHALL consider that the CPM Group Session Data management is supported in this CPM Group Session;
   e. If the SDP offer included in the a=accept-wrapped-types attribute the value application/conference-info+xml and if the CPM Client supports Participant Information Notifications via Conference Information Object, then the CPM Client SHALL consider that Participant Information Notifications via Conference Information Object is applicable in this CPM Group Session;
8. SHALL generate a SIP 200 “OK” response to the SIP INVITE request according to the rules and procedures of [RFC3261], with the following clarifications:
   a. The CPM Client SHALL include an answer SDP as described in section 5.2.1.3 “*SDP Handling at Terminating Nodes*”, with the following clarification:
      i. If the CPM Group Session Data management is supported in this CPM Group Session and the CPM Client also supports the CPM Group Session Data management, then it SHALL include in the a=accept-wrapped-types attribute the value of the CPM Group Session Data Content-Type as specified in section 6.8 “*CPM Group Session Data Management*” and the value multipart/related;
      ii. If Participant Information Notifications via Conference Information Object is supported, it SHALL include in the a=accept-wrapped-types attribute the value application/conference-info+xml, and the value multipart/related (if not already present);
   b. The CPM Client MAY include a display name as specified in section 6.3 “*Display Name*”;
   c. The CPM Client SHALL include the Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID header fields and values as received in the SIP INVITE request;
   d. SHALL include the CPM Address of the CPM User as the preferred originator's CPM Address as described in section 6.1 “*Authenticated Originator’s CPM Address*”;
   e. SHALL include a Session-Expires header field in the SIP 200 "OK" response to the SIP INVITE request with the refresher parameter set based on the value received in the SIP INVITE request and SHALL start the SIP session timer according to the rules and procedures specified in [RFC4028];
   f. SHALL include the option tag 'timer' in the Supported header field and in the Required header field;
9. SHALL send the SIP 200 “OK” response according to the rules and procedures of the SIP/IP core;

Upon receiving a SIP ACK request, the CPM Client:

1. SHALL handle the SIP ACK request according to the rules and procedures of [RFC3261]; and
2. SHALL initiate the Media Plane as in section 7.3.9 “*Media Plane Handling for CPM Sessions*”.
3. If a Session-Replaces header field is included in the SIP INVITE request, the CPM Client SHALL send a SIP BYE request to end the CPM 1-1 Session identified in the Session-Replaces header field.

If the received SIP INVITE was for a CPM Group Session:

a) once the SIP session is successfully established (i.e. a SIP 200 “OK” response was returned) and if Participant Information Notifications via Conference Information Object is not applicable in this CPM Group Session, the CPM Client SHALL subscribe to the conference event package as described in section 7.3.10.1 “*Subscribe to Receiving CPM Group Session Participant Information*” and SHALL maintain the subscription throughout the CPM Client’s participation in the CPM Group Session;

b) Once a SIP NOTIFY containing Participant Information was received as described in section 7.3.10.2 *“Receive Participant Information Notification*”, the CPM Client SHOULD consider that the CPM Group Session to which the CPM User was invited is currently active;

c) Otherwise, if a SIP 404 “Not Found” error response is received for the SIP SUBSCRIBE request, the CPM Client SHALL consider that the CPM Group Session is now inactive. The CPM Client SHALL consider the last Participant Information received (e.g. via the recipients-list in the SIP INVITE, or via the Group State Object or Conference Information Object delivered using MSRP in the CPM Group Session) to contain the last list of Participants left in the CPM Group Session before it became inactive. The CPM Client SHALL use the last Participants list for rendering purposes to inform the CPM User of the Participants left in the CPM Group Session.

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

When a Participant in a CPM 1-1 Session wants to add one or more users to the session, the CPM Client SHALL check that the number of Invited CPM Users on the URI-list does not exceed the maximum number of Participants allowed in a CPM Group Session for a CPM Ad-hoc Group as configured for the CPM Client. If exceeded, the CPM Client SHALL notify the CPM User.

Otherwise, the CPM Client SHALL generate an initial SIP INVITE request according to the rules and procedures of [RFC3261]. In this SIP INVITE request the CPM Client:

1. SHALL set the Request-URI of the SIP INVITE request to the URI of the Controlling Function provisioned in the CPM Client;
2. SHALL include an Accept-Contact header field with the CPM feature-tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’, 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;
3. SHALL include the CPM feature-tag ‘urn:urn-7:3gpp-service.ims.‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ in the Contact header field 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;
4. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group’;
5. SHALL include the CPM Address of the CPM User as the preferred originator's CPM Address, as described in section 6.1 “*Authenticated Originator’s CPM Address*”;
6. SHALL include the CPM release version in the User-Agent header field as specified in Appendix D “*Release Version in User-agent and Server headers*”;
7. If anonymity is requested, SHALL include value "id" in a Privacy header field according to the rules and procedures of [RFC3325];
8. SHALL add the invited user(s) in a MIME resource-list body according to [RFC5366], including also the GRUU of the original invited user that was included in the Contact header field received from that CPM User in the CPM 1-1 Session establishment;
9. SHALL for the originally invited user identity in the MIME resource list, include the Session-Replaces header field as defined in Appendix C “CPM-defined SIP Headers” with the Contribution-ID corresponding to the original session.
10. SHALL insert in the SIP INVITE request a Content-Type header field with multipart/mixed as specified in [RFC2046];
11. SHALL include the Conversation-ID of the ongoing CPM Session and the InReplyTo-Contribution-ID header field used in the initial SIP INVITE request for the CPM 1-1 session and a newly generated Contribution-ID value;
12. SHALL include in the SIP INVITE request a MIME SDP body as a SDP offer as described in section 5.2.1.1 “*SDP Contents when Initiating or Modifying a CPM Session*”. The Media Streams in the SDP offer SHALL be identical to those in the CPM 1-1 Session, with the following differences:
    a. If the CPM Group Session Data management is supported by the CPM Client, it SHALL include in the a=accept-wrapped-types attribute the value of the CPM Group Session Data Content-Type, as specified in section 6.8 “*CPM Group Session Data Management*” and the value “multipart/related”;
    b. If Participant Information Notifications via Conference Information Object is supported, it SHALL include in the a=accept-wrapped-types attribute the value “application/conference-info+xml”, and the value “multipart/related” (if not already present);

13. SHALL send the SIP INVITE request according to the rules and procedures of SIP/IP core.

On receiving a SIP 200 "OK" response to the SIP INVITE request, the CPM Client SHALL handle the response according to the rules and procedures of [RFC3261], with the following clarifications:

1. The CPM Client SHALL generate and send a SIP ACK request as an acknowledgement of the final response towards the CPM Controlling Function;
2. The CPM Client SHALL store the contents of the Contact header field as the CPM Group Session Identity, as described in [RFC4579];
3. The CPM Client SHALL start the SIP Session timer using the value received in the Session-Expires header field according to the rules and procedures of [RFC4028];
4. If the SDP answer included the a=accept-wrapped-types attribute set to the value of the CPM Group Session Data Content-Type, as specified in section 6.8 "*CPM Group Session Data Management*", and the CPM Client supports also the CPM Group Session Data management , then CPM Client SHALL consider that the CPM Group Session Data management is supported in this CPM Group Session;
5. If the SDP answer included in the a=accept-wrapped-types attribute the value "application/conference-info+xml" and if the CPM Client supports Participant Information Notifications via Conference Information Object, then the CPM Client SHALL consider that Participant Information Notifications via Conference Information Object is applicable in this CPM Group Session;
6. If Participant Information Notifications via Conference Information Object is not applicable in this CPM Group Session, then the CPM Client SHALL subscribe to the conference state event package as specified in section 7.3.10 "*Participant Information*" and SHALL maintain the subscription throughout the CPM Client's participation in the CPM Group Session.

7.3.4 Closing a CPM Session

7.3.4.1 Closing a CPM 1-1 Session

To closea CPM 1-1 Session, the CPM Client:

1. SHALL generate a SIP BYE request according to the rules and procedures of [RFC3261], with the Reason Header field as defined in [RFC3326] has a protocol-value set to SIP and a protocol cause=200;
2. SHALL send the SIP BYE request according to the rules and procedures of the SIP/IP core;
3. SHALL release all Media Plane resources corresponding to the CPM Session being closed.

7.3.4.2 Leaving a CPM Group Session

When a Participant wants to leave a CPM Group Session the CPM Client:

1. SHALL generate a SIP BYE request according to the rules and procedures of [RFC3261], with the Reason Header field as defined in [RFC3326] has a protocol-value set to SIP and a protocol cause=200;
2. SHALL set the Request-URI to the CPM Group Session Identity of the CPM Group Session to leave;
3. If anonymity is requested, SHALL include value "id" in the Privacy header field according to the rules and procedures of [RFC3325];
4. SHALL send the SIP BYE request according to the rules and procedures of SIP/IP core;
5. SHALL release all Media Plane resources corresponding to the CPM Group Session being closed.

7.3.4.3 Receiving a CPM Session Closing Request

Upon receiving a SIP BYE request, the CPM Client:

1. SHALL generate a SIP 200 "OK" response according to the rules and procedures of [RFC3261];
2. SHALL send the SIP 200 "OK" response according to the rules and procedures of the SIP/IP core;
3. SHALL release all Media Plane resources corresponding to the CPM Session being closed;
4. If the received SIP BYE request includes a Reason header field with the protocol set to SIP and the protocol-cause set to 200 along with a reason text (e.g. *SIP;cause=200;text="*Call completed elsewhere") as described in [RFC3326], the CPM Client MAY indicate to the CPM User that the CPM Session was established elsewhere (on another device).

5. If the received SIP BYE request for a CPM Group Session includes a Reason header field with the protocol set to SIP and the protocol-cause set to 410 (e.g. *SIP;cause=410;text="*Gone"), the CPM Client MAY indicate to the CPM User that the CPM User was removed from the CPM Group Session. If a Referred-by header is present in the SIP BYE request, the CPM Client MAY use the contained address to indicate the identity of the Participant who initiated the removal request.

The CPM Client SHALL terminate the CPM Session.

7.3.4.4 Receiving a CPM Session Cancellation

Upon receiving a SIP CANCEL request, the CPM Client SHALL follow the actions according to the rules and procedures of [RFC3261];

1. If the received SIP CANCEL request includes a Reason Header field with protocol set to SIP and protocol cause set to 200 and a reason text set to e.g. 'Call completed elsewhere' as described in [RFC3326], the CPM Client MAY indicate to the CPM User that the CPM Session was established elsewhere (on another device).
2. The CPM Client SHALL proceed with termination of the CPM Session.

7.3.5 Invite other Principals to existing CPM Group Session

To add one or more Principals to an ongoing CPM Group Session, the CPM Client SHALL generate a SIP REFER request according to the rules and procedures of [RFC3515] and [RFC7647]. In this SIP REFER request, the CPM Client:

1. SHALL set the Request-URI of the SIP REFER request to the CPM Group Session Identity of an ongoing CPM Session;
2. SHALL ensure this SIP REFER request is sent within the scope of the dialog created with the SIP INVITE request for the ongoing CPM Group Session as per [RFC3515] and [RFC7647];
3. SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session', field 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;
4. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group';
5. SHALL include a Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' field 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;];
6. SHALL include the CPM Address of the CPM User as the preferred originator's CPM Address, , as described in section 6.1 "*Authenticated Originator's CPM Address*";
7. SHALL include a User-Agent header field to indicate the CPM release version as specified in Appendix D "*Release Version in User-agent and Server headers*";
8. If anonymity is requested, SHALL include value "id" in a Privacy header field according to the rules and procedures of [RFC3325];

   NOTE: If anonymity is not allowed for the CPM Group indicated with the Request-URI of the SIP REFER request based on rules specified in the [OMA-XDM-Group] the CPM Session will not be allowed by the CPM Controlling Function hosting the CPM Group.

9. If only one Principal is invited SHALL set the Refer-To header field of the SIP REFER request to the address of the Principal to be added according to the rules and procedures of [RFC3515] and [RFC7647];
10. If more, than one Principal is invited:
    a. SHALL include a Refer-To header field with a pointer (Content-ID Uniform Resource Locator (URL) as per [RFC2392]) that points to the body part that carries the MIME resource-list body according to the rules and procedures of [RFC5368]. The MIME resource-list body SHALL include the listing of address URIs of the Participants to be added;
    b. SHALL include the Content-ID header field set to the value of the CID pointer, with the MIME resource-lists body with the list addresses of the Principals to be added, according to the rules and procedures of [RFC5368];
    c. SHALL include the "multiple-refer" option tag in the Require header field of the SIP REFER request according to the rules and procedures of [RFC4488] and [RFC5368].

11. SHALL set the “method” parameter of the Refer-To header field to “INVITE” according to the rules and procedures of [RFC3515] and [RFC7647];
12. SHALL set the Refer-Sub header field to “false” according to the rules and procedures of [RFC4488] and [RFC5368];
13. SHALL include the “norefersub” option tag in the Require header field of the SIP REFER request according to the rules and procedures of [RFC4488];
14. If the CPM Client knows the number of Participants in the CPM Group Session via a subscription to Participant Information, it SHALL check that the number added Principals plus the number of the Principals already participating to the CPM Group Session does not exceed the maximum number of Participants allowed in an CPM Group Session for a CPM Ad-hoc Group, as provisioned to the device. If exceeded, the CPM Client SHALL notify the CPM User.

    Otherwise, continue with the rest of the steps;
15. SHALL include the Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID header fields set to the values of the ongoing CPM Session;
16. SHALL send the SIP REFER request according to the rules and procedures of the SIP/IP core.

The response to the SIP REFER request SHALL be handled according to the rules and procedures of [RFC7647] and the rules and procedures of the SIP/IP core.

If the CPM Client is aware that the CPM Group Session is a CPM Closed Group Session, then it SHALL not allow the addition of any new participants to the CPM Closed Group Session by the CPM User.

The CPM Client SHALL handle the SIP responses to the SIP REFER according to [RFC7647].

7.3.6 Remove Participants from a CPM Group Session

To remove one or more Participants from an ongoing CPM Group Session, the CPM Client SHALL generate a SIP REFER request according to the rules and procedures of [RFC3515] and [RFC7647]. In this SIP REFER request, the CPM Client:

1. SHALL set the Request-URI of the SIP REFER request to the CPM Group Session Identity of an ongoing CPM Session;
2. SHALL ensure this SIP REFER request is sent within the scope of the dialog created with the SIP INVITE request for the on-going CPM Group Session as per [RFC3515] and [RFC7647];
3. SHALL include an Accept-Contact header field with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’, 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;
4. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group’;
5. SHALL include a Contact header field with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’, 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;
6. SHALL include a User-Agent header field to indicate the CPM release version as specified in Appendix D “*Release Version in User-agent and Server headers*”;
7. If anonymity is requested, SHALL include value “id” in a Privacy header field according to the rules and procedures of [RFC3325];
8. If only one Participant is removed:
    a. If the Participant to be removed is not an anonymous one, set the Refer-To header field of the SIP REFER request to the CPM Address of the removed Participant according to the rules and procedures of [RFC3515] and [RFC7647];
    b. Otherwise, if the Participant to be removed is anonymous, set the Refer-To header field of the SIP REFER request to the Anonymous CPM Address of the removed Participant.
9. If more than one Participant is removed:
    a. SHALL include a Refer-To header field with a pointer (Content-ID Uniform Resource Locator (URL) as per [RFC2392]) to an URI-list in a body part containing the MIME resource-list body according to the rules

and procedures of [RFC5368]. The MIME resource-list body SHALL include the listing of address URIs of the Participants to be removed;

b. SHALL include the Content-ID header field set to the value of the CID pointer, with the MIME <resource-lists> body with the list of the CPM Users to be removed, according to the rules and procedures of [RFC5368];

c. if a Participant to be removed is anonymous then the anonymous CPM Address SHALL be used in the MIME resource-lists;

d. SHALL include the "multiple-refer" option tag to the Require header field according to the rules and procedures of [RFC4488] and [RFC5368].

10. SHALL set the "method" parameter of the Refer-To header field to "BYE" according to the rules and procedures of [RFC3515] and [RFC7647];
11. SHALL set the Refer-Sub header field to "false" according to the rules and procedures of [RFC4488];
12. SHALL include the "norefersub" option-tag in the Require header field of the REFER according to the rules and procedures of [RFC4488];
13. SHALL include the Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID header fields set to the values of the ongoing CPM Session;
14. SHALL send the SIP REFER request according to the rules and procedures of the SIP/IP core.

The response to the SIP REFER request SHALL be handled according to the rules and procedures of [RFC7647] and the rules and procedures of the SIP/IP core.

7.3.7 Modifying a CPM Session

The CPM Client SHALL generate a SIP re-INVITE request according to section 7.3.1 "*Initiating New CPM Sessions*" with the following additional clarifications:

1. The CPM Client SHALL set the Request-URI to the address in Contact header field of SIP INVITE request or SIP 200 "OK" response received during the session establishment;
2. The CPM Client SHALL set each of To header field, From header field and Call-ID header field to the same value as used at the session establishment;
3. The CPM Client SHALL include a modified MIME SDP body as a SDP offer as described in section 5.2.1.1 "*SDP Contents when Initiating or Modifying a CPM Session*";
4. The CPM Client SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

On receipt of the SIP 200 "OK" response to the SIP re-INVITE request the CPM Client SHALL handle the response according to the rules and procedures of [RFC3261], with the following clarifications:

1. The CPM Client SHALL generate and send a SIP ACK request according to the rules and procedures of [RFC3261] and the SIP/IP core;
2. The CPM Client SHALL modify the Media Plane according to the re-negotiated SDP.

7.3.8 Handling a Received CPM Session Modification Request

Upon receiving a SIP re-INVITE request with a MIME SDP body including a new SDP offer, the CPM Client:

1. SHALL check whether the session to be modified by the received SIP re-INVITE exists. If the session does not exist the CPM Client SHALL reject the request with a SIP 481 "Call/Transaction Does Not Exist" response and SHALL include a SIP Warning header field with the warning text set to "123 Session does not exist" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
2. SHALL validate the media parameters and content types listed in the "accept-types" and, if present, in the accept-wrapped-types (if it carries a value other than "*") SDP attributes. If none of the media parameters are acceptable, the CPM Client SHALL reject the request with a SIP 488 "Not Acceptable Here" response and exit this procedure.

   Otherwise, the CPM Client SHALL select a subset of the acceptable received media parameters and content types, to replace the original SDP parameters, and continue with the rest of steps;
3. SHALL generate a SIP 200 "OK" response according to the rules and procedures of [RFC3261];

4. SHALL include a Session-Expires header field in the SIP 200 "OK" response to the SIP re-INVITE request with the refresher parameter set based on the value received in the SIP re-INVITE request and SHALL re-start the SIP session timer according to the rules and procedures specified in [RFC4028];
5. SHALL include the option tag 'timer' in the Supported header field and in the Required header field;
6. SHALL include a MIME SDP body as the SDP answer as described in section 5.2.1.3 "*SDP Handling at Terminating Nodes*";
7. SHALL include the Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID header fields and values received in the SIP INVITE request;
8. SHALL send the SIP 200 "OK" response according to the rules and procedures of the SIP/IP core.

Upon receiving a SIP ACK request, the CPM Client:

1. SHALL handle the SIP ACK request according to the rules and procedures of [RFC3261];
2. SHALL change media parameters as specified in [RFC4975] for MSRP based media streams and, respectively, [RFC3550] for RTP/RTCP-based Media Streams.

7.3.9 Media Plane Handling for CPM Sessions

CPM Clients MAY support different text and media types (including SMIL) as defined in [3GPP TS 26.141].

When inviting, being invited or joining a CPM Session, a CPM Client SHALL negotiate the media parameters with the invited CPM Client or the CPM Participating and Controlling Function by using SDP within SIP communication.

7.3.9.1 MSRP-based Media Streams

For MSRP-based Media Streams, the Client SHALL follow the rules and procedures defined in [RFC4975] and in [RFC6714] for the establishment of sessions and the exchange of CPM Chat Messages, with the following clarification:

1. When sending a CPM Chat Message, the CPM Client SHALL include the content in a CPIM wrapper and set the "To" header field of the CPIM message as follows.
    a. If the CPM Chat Message is to be sent to one CPM user or to one non-CPM Principal within a CPM Group Session, the CPM Client SHALL set the "To" header field of the CPIM message to the address of the recipient. This applies for both CPM Chat Messages, "isComposing" notifications and for sending IMDN via MSRP during the CPM Group Session;
    b. If the CPM Chat Message is to be sent to a CPM Ad-hoc Group, the CPM Client SHALL set the "To" header field of the CPIM message to the address of the CPM Controlling Function of the on-going CPM Group Session, or to the anonymous URI;
    c. If the CPM Chat Message is to be sent to a CPM Pre-defined Group, the CPM Client SHALL set the "To" header field of the CPIM message to the address of the target CPM Pre-defined Group;
2. When sending a CPM Chat Message, the CPM Client SHALL set:
    a. the Content-Type header field of the MSRP message to message/cpim, as defined in [RFC3862];
    b. the Content-Type header field of the CPIM Message to the Content-Type of the payload found inside the CPIM wrapper.
    c. if the CPIM wrapped message body includes a multipart MIME (e.g. multipart/related, multipart/mixed), then:
        i. SHALL add a CPIM cpm.Payload-Type header field, set to the list of Content-Types included in the payload of the body of the message.
    d. the "From" header field of the CPIM Message to
        i. either the anonymous URI or the CPM User's Authenticated Originator's CPM Address in a CPM 1-to-1 Session
        ii. To the CPM User's Authenticated Originator's CPM Address in a CPM Group Session including the CPM user's Display Name as Formal Name if available and if the user isn't participating in the session anonymously (see section 6.3 "Display Name and Anonymity).
3. When receiving a CPM Message, the CPM Client SHALL handle the CPIM message:
    a. as described in section 7.2.2.3 "*CPIM Handling of Received CPM Standalone Messages and SIP IMDNs*", to check and handle content indirection cases or any IMDN content;

   b. otherwise, for any other content it SHALL process based on the values of the Content-Type headers received.
      i. If the CPM Client does not support a received Content-Type, it SHALL return an MSRP error response “415”.
      ii. Otherwise, it SHALL process any received content that it understands according to its Content-Type and respond with MSRP 200 “OK”.

4. The CPM Client SHALL request delivery and interworking reports and MAY request read reports for all CPM Chat Messages using IMDN as described in sect. 5.4 “*Disposition Notifications*”;
5. The CPM Client SHALL send any “isComposing” messages via MSRP as described in section 5.5 “*"isComposing" notifications*”. In a CPM 1-1 Session the “isComposing” messages shall be sent without a CPIM wrapper;
6. When sending a CPM event, the CPM Client SHALL ensure that only events related to the CPM Session are transmitted;
7. When receiving a CPM event, the CPM Client SHALL process the CPM event accordingly and SHALL send a response back if one is required as specified in section 6.7.3 “*Handling of the CPM Event Reporting requests and responses*”;
8. The client SHALL take into account the maximum chunk size negotiated according to section 5.2.1, if any;
9. If the CPM Group Session Data management is supported in this CPM Group Session and the CPM Client has to
   a. set or delete the subject, bulletin or the icon or
   b. has to set, delete or move a role of a participant,

   then the CPM Client SHALL include in the body of the MSRP SEND a group-data request as specified in section 6.8 “*CPM Group Session Data Management*”. On reception of a MSRP SEND with a group-data response as specified in section 6.8 “*CPM Group Session Data Management*” the CPM Client SHALL evaluate the response-code value and inform the CPM User about the outcome of the request;
10. When receiving a MSRP SEND request with a body with the Content-Type of “application/conference-info+xml”, then the CPM Client
    a. SHALL handle the body according to the rules and procedures of [RFC4575], supporting the rules described in section 9.2.14.3 “*Sending Participant Information Notifications* ”;
    b. if CPM Group Session Data Management is supported, SHALL check if the <subject> element is present and if the value is different than the current value of the CPM Group Session subject that the CPM Client has for that CPM Group Session, it SHALL replace it locally with the newly received value;
    c. if CPM Group Session Data Management is supported, SHALL check if an element <icon> is present and if so, it SHALL check if the value is different than the current value of the CPM Group Session icon that the CPM Client has for that CPM Group Session. If different, the CPM Client SHALL replace it locally with the newly received value.

7.3.9.2 RTP/RTCP-based Media Streams

For RTP/RTCP-based Media Streams, the CPM Client SHALL follow the rules and procedures defined in [RFC3550] for the establishment of sessions and the exchange of Media.

The CPM Client SHALL establish a RTP session according to the rules and procedures of [RFC3550]. If multiple media are used, then each of them is transmitted as a separate RTP session. That is, separate RTP and RTCP packets are transmitted for each medium using two different UDP port pairs and/or multicast addresses.

When the SIP/IP core corresponds with 3GPP/3GPP2 IMS, the procedures described in [3GPP TS 24.229] / [3GPP2 X.S0013.004] and described in [GSMA IR.92] apply for handling RTP/RTCP sessions.

7.3.10 Participant Information

7.3.10.1 Subscribe to Receiving CPM Group Session Participant Information

When a CPM Client needs to subscribe to CPM Group Session Participant Information, the CPM Client:

1. SHALL generate a SIP SUBSCRIBE request, according to the rules and procedures of [RFC6665] and [RFC4575];
2. SHALL set the Request-URI of the SIP SUBSCRIBE request to the CPM Group Session Identity;

3. SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' 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;
4. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group';
5. SHALL include a Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' 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;
6. SHALL include the CPM Address of the CPM User as the preferred originator's CPM Address, as described in section 6.1 "*Authenticated Originator's CPM Address*";
7. SHALL send the SIP SUBSCRIBE request according to the rules and procedures of the SIP/IP core.

The response to the SIP SUBSCRIBE request SHALL be handled according to the rules and procedures of [RFC6665] and [RFC4575], and rules and procedures of the SIP/IP core.

When needed the CPM Client SHALL terminate the subscription according to the rules and procedures of [RFC6665].

7.3.10.2 Receive Participant Information Notification

Upon receiving an incoming SIP NOTIFY request that is part of the same SIP dialog as a previously sent SIP SUBSCRIBE request for subscribing to participation information, the CPM Client:

1. SHALL handle the request according to the rules and procedures of [RFC6665] and [RFC4575], supporting the <status> element values and <disconnection-method> element values described in sect. 9.2.14.3 "*Sending Participant Information Notifications* ";
2. MAY display the Participants of the CPM Group Session by that are included in the chat Blacklist URI-list stored in [OMA-XDM-List];
3. SHALL check if the <subject> element is present and if the value is different than the current value of the CPM Group Session subject that the CPM Client has for that CPM Group Session, it SHALL replace it locally with the newly received value, and the <subject-ext> element data;
4. SHALL check if an element <icon>, as defined in section 6.8 "*CPM Group Session Data Management*" and Appendix P. is present and if so, it SHALL check if the value is different than the current value of the CPM Group Session icon that the CPM Client has for that CPM Group Session. If different, the CPM Client SHALL replace it locally with the newly received value;
5. SHALL check if the <bulletin> element is present and if the value is different than the current value of the CPM Group Session bulletin that the CPM Client has for that CPM Group Session, it SHALL replace it locally with the newly received value.

7.3.11 Handling Deferred CPM 1-1 Session Message Delivery

If any 1-1 CPM Chat Message(s) is pending deferred delivery to the CPM Client, the SIP INVITE request is received from the CPM Participating Function as described in section 8.3.2.9.1.1 "*1-1 CPM Session delivery*".

When the CPM Client receives an initial SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' included in the Accept-Contact header field, the 'From' header field value of CPMDeferredMgmtURI and a Referred-By header field including the originator's address, the CPM Client SHALL follow the procedures described in section 7.3.2 "*Receiving a CPM Session Invitation*" with the following clarifications:

1) SHALL use the CPM Session established between Participating Function and CPM Client to receive the deferred delivery of CPM Chat Messages (chat messages) and to send any IMDN notifications back. The CPM Client SHALL NOT allow the CPM User to send any messages back into the deferred delivery session;
2) If the CPM Client was requested to send back disposition notifications for any one of the deferred delivery chat messages and

   the CPM User actions triggered sending of IMDN notifications (e.g., display notifications) after the deferred delivery session ended, the CPM Client SHALL send each disposition notifications using a SIP MESSAGE as described in section 5.4 "*Disposition Notifications*".
3) CPM Client SHALL use Conversation-ID to determine whether corresponding CPM 1-1 Session is known to the CPM Client (for rendering purposes). If the CPM User accepts the delivery (e.g. auto-accept option on device, or manual acceptance), the CPM Client SHALL respond with a 200 OK to accept the delivery.

4) If the CPM User does not wish to receive deferred CPM Message at this time (e.g. CPM User is roaming, etc), the CPM Client SHALL use error code 480 "Temporarily not available" to indicate the preference to receive the deferred CPM Message at a later time.

NOTE: If CPM User replies with a new message to any of the newly delivered deferred messages, then CPM Client SHALL initiate a new CPM 1-1 Session as described in section 7.3.1.1 "*Initiating a CPM 1-1 Session*".

7.3.12 Handling Deferred CPM Group Session Message Delivery

CPM Client SHALL support deferred message delivery of a missed CPM Group Session or missed parts of a CPM Group Session. The deferred message delivery of the CPM Group Session is performed via a CPM Group Session established with the CPM Client. The CPM Group Session can be established in the following ways:

1) By receiving SIP INVITE request from CPM Participating Function to invite the CPM Client back into the CPM Group Session. This CPM Group Session invitation request triggered for deferred delivery is identical to the CPM Group Session request that originally invited the CPM User to join the CPM Group Session. The CPM Client SHALL handle it as described in section 7.3.2 "*Receiving a CPM Session Invitation*";
2) By CPM Client initiating a SIP INVITE to re-join the CPM Group Session, after the CPM Client came back online and has successfully registered in IMS:
    a. the CPM Client may re-join the CPM Group Session, following procedures as described in section 7.3.1.5 "*Re-joining a CPM Long-lived Group Session*".

    NOTE: The CPM Participating Function will attempt inviting the CPM Client either to deliver any deferred CPM Chat Messages for the CPM Group Session, or to re-connect the CPM Client into the CPM Group Session if it is still active. The CPM Client SHOULD NOT attempt re-joining the previously CPM Group Session, unless CPM User has CPM Message to send.

7.4 CPM File Transfer

7.4.1 CPM File Transfer Session Initiation

When a CPM Client needs to send or receive one or more files, the CPM Client:

1. SHALL follow the procedures defined in 7.3.1.1 "*Initiating a CPM 1-1 Session*" if the CPM File Transfer is to one recipient or in 7.3.1.2 "*Initiating a CPM Group Session for a CPM Ad-hoc Group*" if the CPM File Transfer is to a list of recipients, or in 7.3.1.3 "*Initiating a CPM Group Session for a CPM Pre-defined Group*" if the CPM File Transfer is to a pre-defined group and SHALL follow the rules and procedures of [RFC5547] with the following clarifications:
    a. SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer' 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, instead of the 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' used in the above referenced CPM procedures;
    b. SHALL include a Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer' 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 instead of the 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' used in the above referenced CPM procedures;
    c. SHALL set the P-Preferred-Service header field with the value of the CPM Feature Tag as follows:
        i. If the CPM File Transfer is sent to one recipient, the ICSI SHALL be set to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer'
        ii. Otherwis, if the CPM File Transfer is sent to a group of reciients, the ICSI SHALL be set to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer.group';
    d. SHALL handle the SIP session timer parameters defined in [RFC4028], as specified in section 7.2.1.2 "*Sending a Large Message Mode CPM Standalone Message*";
    e. SHALL add the relevant media attributes to the SDP as specified in [RFC5547], with the following additional considerations:
        i. If the request is to send (push) a file, the "name", "size" and "type" sub-attributes of the "file-selector" attribute SHALL be included;

        ii. If the request is to receive (pull) a file, as many of the "file-selector" sub-attributes as are known SHALL be included;

    f. In the case of a CPM File Transfer to a single recipient and more than one file is to be sent, each set of file descriptors SHALL be sent in separate m-lines as described in [RFC5547];
    g. If the request is to send one or more files, the CPM Client SHALL validate that the size of a file being sent does not exceed the provisioned Max File Transfer Size value before continuing with the above procedures.
    h. If the recipient's capability indicates thumbnail support and the sending CPM Client supports it as well, the sending CPM Client SHALL include thumbnail information in the SDP of the SIP INVITE request by using the file-icon attribute as defined in [RFC5547], and include the file thumbnail in the body of the SIP INVITE.
    i. When requesting a disposition notification, the CPM Client SHALL set the disposition notification according to the rules and procedures as described in section 5.4 "*Disposition Notification*", where the CPIM body including IMDN header fields is included in the SIP INVITE body.

    NOTE: the method to determine the thumbnail capability by the CPM Clients before initiating the CPM File Transfer request is out of scope of this specification.

    NOTE: If there is more than one recipient transfer of multiple files is not supported.

2. When the MSRP 200 "OK" response for the last MSRP SEND is received, or when an MSRP REPORT is received when a success report was requested:
    a. The CPM Client MAY check if there are other files to be transferred to the same destination;
    b. If there are additional files to be transferred and the CPM Client wants to reuse m-lines for transferring them as described in [RFC5547], the CPM Client SHALL send re-INVITE with the following clarifications:
        i. The CPM Client SHALL reuse the needed m lines by creating new relevant attributes for the new files according to procedures of [RFC5547];
        ii. if more m lines were used and are no longer needed, the CPM Client SHALL set the port in the m-line to zero (i.e. m=0) to close the MSRP session for the particular files that has been transferred according to procedures of [RFC5547];
    c. Otherwise, the CPM Client SHALL follow the procedures defined in 7.4.3 "*CPM File Transfer Session Release*";
3. If an MSRP error response or an MSRP REPORT with a failure report is received, the CPM Client SHALL indicate to the CPM User that the CPM File Transfer has failed.

7.4.1.1 CPM File Transfer Session Initiation in CPM Session

When a CPM Client needs to send files to the CPM Participant in an active CPM Session, the CPM Client SHALL follow the procedures in 7.4.1 "*CPM File Transfer Session Initiation*" with the following clarifications:

1. If within a CPM Session, the CPM Client SHALL use the Conversation-ID and Contribution-ID of the associated CPM Session.
2. If within an active CPM Group Session, the CPM Client:
    a. SHALL set the Requested-URI address to the CPM Group Session Identity of the associated CPM Group Session;
    b. SHALL not include the recipient-list;
    c. If the disposition notification is requested from each CPM Participant for the CPM File Transfer, then the CPM Client SHALL include the IMDN CPIM wrapper with an empty CPIM body and it SHALL indicate support also for IMDN content-type by adding the value "*message/imdn+xml"* in the SDP a=accept-wrapped-types attribute.
3. If the CPM File Transfer is sent within an inactive CPM Group Session, the CPM Client SHALL restart the CPM Group Session as described in 7.3.1.5 "*Re-joining CPM Long-lived Group Session*". Then the CPM Client SHALL initiate the CPM File Transfer in an active CPM Group Session as described in Step 2 above.

7.4.2 Receiving a CPM File Transfer Request

Upon receiving a SIP INVITE request containing a CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer’ included in the Accept-Contact header field, the CPM Client:

1. SHOULD render to the CPM User the file description parameters received as well as the identity of the original sender of the request if available;
2. SHALL accept from the CPM User input regarding the file(s):
   a. which he is willing to accept in case of a push request, or,
   b. which he is willing to send, in the case of a pull request;
3. SHALL inspect the SDP attributes relating to the file(s) to be transferred, and if the request is for receiving one or more files, the CPM Client shall validate whether the “file-selector:size” attribute exceeds the provisioned value for this CPM Client.
4. If the CPM User accepts at least one file, SHALL return a SIP 200 “OK” response :
   a. including the SIP session timer parameters defined in [RFC4028], as specified in section 7.2.2.2 “*Receiving a Large Message Mode CPM Standalone Message*”;
   b. with an SDP answer containing the file identities of the files as described in [RFC5547], and otherwise SHALL return a 603 “Decline” response.
5. SHALL execute the Media Plane procedures as described in section 7.3.9.1 “*MSRP-based Media Streams*” to receive the files.
6. After completing the MSRP procedures for each transferred file, the CPM Client SHALL check whether the CPM File Transfer Request contains the request for disposition notification in the body of the SIP INVITE. If true, the CPM Client SHALL send the appropriate disposition notification as described in Section 5.4.1 “*Generate Delivery Notification*” and Section 2 “*Generate Read Report*” via SIP MESSAGE.

7.4.3 CPM File Transfer Session Release

When a CPM File Transfer session is completed:

- the sending CPM Client SHALL send a SIP BYE to close the session as described in section 7.3.4.1 “*Closing a CPM 1-1 Session*” , with the following clarifications:
  - Upon receiving the 200 “OK” SIP response to the SIP BYE request, if the CPM File Transfer was stored in the CPM Message Store by the CPM Participating Function serving the CPM Client, the CPM Client SHALL retrieve the UID value of the CPM File Transfer object from the Message-UID header field. This value is used further by the CPM Client for correlation during the message synchronization process with CPM Message Store.
- the receiving CPM Client, upon receiving a SIP BYE request to close CPM File Transfer session:
  - if the CPM File Transfer object was stored in the CPM Message Store, the CPM Client SHALL retrieve the UID value of the CPM File Transfer object from the Message-UID header field of the SIP BYE request;
  - If a Message-UID header field is not present in the SIP BYE request, or the SIP BYE request fails to arrive at the CPM Client, the CPM Client SHALL identify when message duplicates exist locally in the client (e.g. received via the SIP delivery and also received via IMAP synchronization) based on:
    - the IMAP folder in which the message is stored (i.e. folder SHALL match the identity of the other party in a 1-1 CPM Standalone Message and the Conversation-ID for the CPM Group Standalone Message) and,
    - the CPM SIP headers fields of a message delivered via SIP match the values of the MIME headers of message objects received during the IMAP synchronization, as described in the table 1 section 7.2.2.1 “*Sending a Pager Mode CPM Standalone Message and SIP IMDNs*”;
  - CPM Client SHALL return a 200 “OK” SIP response to the CPM Participating Function as described in [RFC3261].
  - If a disposition notification is requested, the CPM Client SHALL generate the appropriate disposition notification as described in section 7.4.2 “*Receiving a CPM File Transfer request*” step 6.

If the file sender or the file receiver wishes to abort the file transfer during the CPM File Transfer, the CPM Client SHALL follow section 8.3.5 "*Aborting an Ongoing File Transfer Operation*" of [RFC5547].

The CPM Client determines that a CPM File Transfer interruption was:

a. User triggered, as described in [RFC5547]:

   i. by the sending CPM Client:

      1. the sending CPM Client SHALL abort the MSRP request by including the '#' character in the continuation field of the end-line of the on-going MSRP SEND request, according to the procedures in [RFC4975]. The receiving CPM Client SHALL acknowledge this MSRP SEND request with a MSRP 200 OK response;

      2. if the CPM File Transfer request included other files than the one being interrupted, then the sending CPM Client SHALL update the CPM File Transfer session via a SIP re-INVITE request as described in [RFC5547].

      3. Otherwise, if the aborted file is the only file, or the last file, transmitted within the CPM File Transfer request (SIP INVITE) then the CPM Client SHALL NOT send a re-INVITE with the "m="line set to 0, but it SHALL close the CPM File Transfer session by sending a SIP BYE request with the Reason header field containing a reason-protocol set to SIP and a reason-cause set to value of "200" and SHALL release the Media Plane resources associated with this CPM File Transfer request;

      4. If the interruption by the sending CPM Client was due to receipt of a MSRP failure report or a failure transaction response for a chunk that is currently being delivered, as described in section 7.3.2 of [RFC4975], the sending CPM Client SHALL abort the outgoing MSRP request as per procedures described in [RFC4975] and [RFC5547], by including the '#' character in the continuation field of the end-line of a SEND request. As an addition to [RFC5547] procedures, if there are no more files to be transmitted within the on-going CPM File Transfer request, the sending CPM Client SHALL send a SIP BYE request with a Reason header field including reason-protocol set to SIP and a reason-cause set to "480" to indicate that it was not a user triggered interruption. After that it SHALL release the Media Plane resources associated with this CPM File Transfer request.

   ii. by the recipient CPM Client:

      1. If the value of the MSRP Failure-Report header of the incoming MSRP SEND request was set to "yes", or there was no Failure-Report header present (hence the default value of "yes" is assumed), the recipient CPM Client SHALL respond with a MSRP 413 error response to the last received MSRP SEND request as per procedures in [RFC5547]. The sending CPM Client SHALL stop sending further MSRP SEND requests upon receiving the MSRP 413 error response;

         a. As an addition to [RFC5547] procedures, in case the aborted file is the only file, or the last file, transmitted in the CPM File Transfer request (SIP INVITE), then the CPM Client SHALL NOT send a re-INVITE with the "m="line set to 0 but SHALL rather send a SIP BYE request containing Reason header field with a reason-protocol set to SIP and a reason-cause set to value of "200", to close the CPM File Transfer session and SHALL release the Media Plane resources;

2. If the value of the MSRP Failure-Report header of the incoming MSRP SEND request was set to “no”, the recipient CPM Client SHALL close the MSRP session by sending a new offer via a re-INVITE in which the “m=” line corresponding to the aborted file is set to 0 as per procedures in [RFC5547];

    a. As an addition to [RFC5547] procedures, in case the aborted file is the only file transmitted in the CPM File Transfer request (SIP INVITE), or the last file, then the CPM Client SHALL NOT send a re-INVITE with the “m=”line set to 0 but SHALL rather send a SIP BYE request with Reason header field containing the reason-protocol set to SIP and a reason-cause set to “200” to close the CPM File Transfer session. Upon receiving it, the sending CPM Client SHALL stop sending further MSRP SEND requests and SHALL respond with a SIP 200 OK response to the SIP BYE request, to close the CPM File Transfer session and SHALL release the Media Plane resources;

3. If an ongoing MSRP SEND request is being transmitted by the sending CPM Client when it receives a MSRP 413 error response, or a SIP BYE request, the sending CPM Client SHALL abort the MSRP message by including the '#' character in the continuation field of the end-line of a SEND request, according to the MSRP procedures in [RFC4975].

b. Network triggered:

    1. when the CPM Client receives a SIP BYE request with a Reason header field with the reason-protocol set to SIP and the reason-cause set to a value different than “200”. In this case, the CPM Client SHALL assume that CPM File Transfer has been interrupted by the network, as described in section 8.2.2.3 “*Handle a SIP BYE Request*”.

All SIP BYE requests sent by the CPM Client to terminate a SIP session SHALL be generated as described in section 7.3.1.4 “*Closing a CPM 1-1 Session*” for 1-1 File Transfer and to section 7.3.4.2 “*Leaving a CPM Group Session*” for Group File Transfer.

7.4.4 Fetching a Deferred CPM File Transfer file(s)

When the CPM Client failed to respond to the SIP INVITE within the session expiry timer, the CPM Participating Function cancels the SIP INVITE request towards the receiver by sending a SIP CANCEL, as described in Section 8.3.3.3 “*Handling Deferred CPM File Transfer File(s)*”.

Upon receiving SIP CANCEL from CPM Participating Function, corresponding to the existing CPM File Transfer request, with reason header field, Reason: SIP;cause=408 ;text="User not responding”, the terminating CPM Client SHALL keep metadata of CPM File Transfer request, such as the device identifier of the File Transfer originator, the file-transfer-id and the file-name SDP attributes.

When fetching a deferred CPM File Transfer file(s), the terminating CPM Client SHALL use the previously stored file-transfer-id and file-name SDP attributes to construct a new CPM File Transfer request as defined in 7.4.5.1 “*Requesting Side*” to initiate a new File Transfer.

Upon receiving a 200 OK response, the terminating CPM Client SHALL set up the Media Plane according to [RFC5547].

7.4.5 Resuming an interrupted CPM File Transfer

During the CPM File Transfer session, the CPM Clients SHALL track how much of the data has been sent or received (as applicable). If the File Transfer session is interrupted, the affected CPM Client MAY request to resume the interrupted file transfer from the point of interruption, without having to transfer the entire file again.

7.4.5.1 Requesting Side

The CPM Client which requests to resume a CPM Filer Transfer (the requestor) SHALL follow the procedures defined in 7.4.1 "*CPM File Transfer Session Initiation*" to initiate a new File Transfer and resumption procedures of [RFC5547], with the following clarifications:

1. SHALL include in the SDP appropriate attributes and values according to [RFC5547] to indicate the file(s) for which the CPM File Transfer session activity is to resume, such as:
    a. SHOULD include the 'file-range' attribute in SDP set to the range of octets from the file (start and stop offset octets, both inclusive) as per procedures described in [RRC5547], to indicate to the file sender (e.g. sending CPM Client, CPM Participating Function) where to start the resumption of the file transmission. The absence of the 'file-range' attribute in SDP indicates that the CPM File Transfer SHALL start with the first octet of the file and end with the last octet (i.e., the whole file is transferred);
    b. SHALL contain the following direction attribute:
        i. "a=recvonly" indicating a CPM File Transfer pull request, as per procedures described in [RFC5547], if the resume requestor is the original receiver;
        ii. "a=sendonly" if the resume requestor is the original sender.
2. Upon receiving a 200 OK response, the requestor CPM Client SHALL set up the Media Plane, with the appropriate Byte Range header field values in MSRP request, according to [RFC5547].
3. Once the File Transfer is completed, the file sending CPM Client SHALL terminate the session following same procedures as described in section 7.3.4.1 "*Closing a 1-1 CPM Session*".

7.4.5.2 Responding Side

When receiving a new SIP INVITE for a CPM File Transfer request, the CPM Client respond to a resume request (the responder), SHALL proceed as follows:

1. If the responding CPM Client is the original sender, and still has the requested file(s), it SHALL send back a 200 OK response with the SDP containing the requested file name(s) and the associated 'file-range' attribute in the SDP answer with the same range of values as requested in the SIP INVITE. The device identifier used in the CPIM From header is set as described in section 6.1.1 "*Identifying the sending user and device in SIP requests and responses*".
2. If the responding CPM Client is the original recipient, it SHALL follow the procedure defined in 7.4.2 "*Receiving a CPM File Transfer Request*", with the appropriate Byte-Range header field values in the MSRP request. The receiving CPM Client SHALL send back any requested IMDN via SIP MESSAGE.

If the responding CPM Client:

i. does not support CPM File Transfer resumption, or

ii. if the file for which resumption is requested or offered is not known on the responding CPM Client (e.g. the originally transmitted bytes were discarded, or the file has changed from the version sent originally), then:

   a) if the resumption of more than one file was requested, the file sender SHALL set the port number of their associated "m=" lines to zero as per regular SDP [RFC4566] procedures, in a SIP 200 OK response; or

   b) if resumption of only one file was requested,

then the file sender's CPM Client SHALL return a SIP 488 "Not Acceptable Here" response.

7.5 Handling of Denoted Messages

The CPM Client SHALL support sending and receiving denoted messages in a CPM Group Session. If the CPM user denote his/her message to one specified Participant in the CPM Group, the CPM Client SHALL include a 'cc' header in the CPIM message with the value set to the address of the specified CPM User. If the CPM User denote his/her message to several specified Participants in the CPM Group, the CPM Client SHALL include several 'cc' headers in the CPIM message respectively.

When receiving a denoted message, the CPM Client SHALL check ‘cc’ header(s) in the CPIM message. And the CPM Client SHALL distinctly notify the CPM User if the value of the ‘cc’ header in the received message matches the address of the CPM User .

The mechanism is applicable for both group standalone messages and group session messages.

8. Procedures at CPM Participating Function

8.1 Registration

In order for the CPM Participating Function to obtain:

A) IMS registration information of a CPM Client the CPM Participating Function receives from and/or subscribes to the SIP/IP core as described in this subclause.

   The SIP/IP core sends to the CPM Participating Function all SIP REGISTER requests where at least one of the contacts in the SIP REGISTER request includes one of the CPM Feature Identifiers defined in Appendix H.3 "*Proposed Formats for CPM Feature Identifiers*".

B) When interworking is needed, core network registration information related to the capability of the CPM Client on a Primary Device to receive SMS. The CPM Participating Function SHALL receive and handle interworking event information from the CPM Interworking Function as described in section 8.1.5 "*Core Network Registration Information*".

8.1.1 Receive SIP REGISTER Notification

The CPM Participating Function SHALL support receiving 3rd party REGISTER from the SIP/IP core according to [3GPP TS 24.229] clause 5.4.1.7.

Upon receiving a 3rd party SIP REGISTER request the CPM Participating Function:

1. SHALL generate a SIP 200 "OK" response according to the rules and procedures of [RFC3261]; and,
2. SHALL send the SIP 200 "OK" response according to the rules and procedures of SIP/IP core; and
3. SHALL, if the CPM Participating Function needs information from the SIP REGISTER request sent by the CPM Client (e.g. the User-Agent header field for the purposes of determining whether the CPM Client is suitable as described in section 6.5 "*Communicating with the ISF and IWF"*), extract the required information from a included "message/sip" MIME body carrying a REGISTER request as described in section 5.7.1.1 of [3GPP TS 24.229] if such a body is included;
4. SHALL, if the CPM Participating Function needs information from the response to the SIP REGISTER request provided to the CPM Client (e.g. the CPM Client's GRUU to address the CPM Client as described in section 6.1.2), extract the required information from a included "message/sip" MIME body carrying a SIP response as described in section 5.7.1.1 of [3GPP TS 24.229] if such a body is included;
5. MAY, if the request is determined to be from a CPM Client on a Primary Device, remove any pending messages for deferred delivery to the Primary Device provided that they were delivered to other CPM Clients already, subject to service provider policies.

If the CPM Participating Function is not already subscribed to the "reg" event package as specified in [RFC3680] for the CPM User Address received in the SIP REGISTER request, the CPM Participating Function:

1. SHALL generate a SIP SUBSCRIBE request according to the rules and procedures of [RFC6665] and [RFC3680];
2. SHALL set the Request-URI of the SIP SUBSCRIBE request to the CPM User Address received in the SIP REGISTER request;
3. MAY set an expiration timer in Expires header with a value according to the rules and procedures of [RFC3903], in the same range as the registration timer recommendations for the SIP/IP core;
4. SHALL send the SUBSCRIBE request according to the rules and procedures of the SIP/IP core.

The responses to the SIP SUBSCRIBE request SHALL be handled according to the rules and procedures of [RFC6665] and [RFC3680], and rules and procedures of the SIP/IP core.

8.1.2 Receive Registration Event Information Notifications

Upon receiving a SIP NOTIFY request as the result of the "reg" event subscription, containing the Registration Event Information as specified in [RFC3680] and [RFC5628], the CPM Participating Function:

1. SHALL generate a SIP 200 "OK" response according to the rules and procedures of [RFC6665] and [RFC3680];
2. SHALL send the SIP 200 "OK" response according to the rules and procedures of SIP/IP core;

3. SHALL store the Registration Event Information, if there is no Registration Event Information already associated with the CPM User address. Otherwise, the CPM Participating Function SHALL replace the old Registration Event Information with the Registration Event Information received in the SIP NOTIFY request.

The CPM Participating Function SHOULD determine which CPM Client of a CPM User resides on the Primary Device. The remaining CPM Client(s) therefore are understood to reside on Secondary Device(s).

8.1.3 Terminating the Subscription to Registration Event Information

The CPM Participating Function MAY terminate the subscription at any time, according to the rules and procedures of [RFC6665] and [RFC3680].

8.1.4 Using the Registration Event Information

The Registration Event Information received and stored in the procedure described in section 8.1.2 "*Receive Registration Event Information Notifications*" allows the CPM Participating Function to determine:

1. The registration state of a CPM User; and
2. The registered devices of the CPM User that can be used to deliver the CPM requests and trigger deferred delivery to, and either the GRUU's associated with each of these devices, or the +sip.instance containing an unique instance identifier associated with each of these devices.
3. The device capabilities according to [RFC3840] of each of the devices in Step 2.

Whenever the CPM Participating Function needs to know the registration state or the registered devices of a CPM User, it SHALL use the stored Registration Event Information associated with the CPM address of a CPM User.

NOTE: Only the terminating CPM Participating Function will use this Registration Event Information to decide the handling of incoming CPM Standalone Messages, CPM File Transfers and CPM Session Invitations.

8.1.5 Core Network Registration

Upon deferral of a CPM Message, CPM Session or CPM File Transfer, the CPM Participating Function uses the indication of SMS support of a CPM Client on a Primary Device as a delivery trigger to the CPM Client on a Primary Device.

When the CPM Participating Function receives a SIP MESSAGE request from the CPM Interworking Function containing the following:

A. A service identification for the request, as described in section 6.7.1 "*Service Identification*";
B. a CPM Event Reporting Content-Type in the body, as defined in section 6.7.3 "*CPM Event Reporting Data Format*" containing the interworking event element <cpm-event-iw> where the <ready-for-sms> event is present.

In that case, the CPM Participating Function:

1) SHALL respond with a SIP 200 "OK" to the CPM Interworking Function, and;
2) SHALL extract the Authenticated CPM User's address (e.g. MSISDN) from the <cpm-user-address> element and SHALL determine if there are any pending messages for that CPM User.
    i. If no CPM Messages, CPM Sessions or CPM File Transfers have been deferred for this CPM User, then CPM Participating Function SHALL ignore the interworking event, and SHALL NOT continue with the next steps;
    ii. SHALL wait for a period of time determined by service provider policy before continuing with the next steps;

        NOTE: This delay is included to allow a potential parallel registration over IP access to come in which may void the need to deliver the pending messages through interworking (see section 8.1.1 "*Receive SIP REGISTER Notification*").

    iii. If there are no longer deferred deliveries pending for the CPM User, then the CPM Participating Function SHALL ignore the interworking event and SHALL NOT continue with the next steps;
3) SHALL consider this a valid trigger to initiate the delivery of any pending CPM Messages, or CPM Sessions, to the CPM User, via interworking.
4) The CPM Participating Function SHALL check the service provider policy about handling undelivered messages, as described in section 8.3.6 "*Delivery Policies in Terminating CPM Participating Function*".

8.2 Procedures in the Originating Network

A CPM Participating Function in an originating network handles incoming CPM communications in the following manners:

- Upon receiving a SIP MESSAGE request with the CPM Feature Tag set to one of:
    - ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg’, or
    - ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg’, or
    - ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’, or
    - ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer’,

  as defined in Appendix H included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP MESSAGE request as described in section 8.2.1.1 “*Handle a Pager Mode CPM Standalone Message and SIP IMDNs*”. If any additional strings (e.g. IARIs) are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences) subject to service provider policies.
- Upon receiving a SIP MESSAGE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg’ defined in Appendix H included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP MESSAGE request as described in section 6.7.3.2 “*Receiving One Time CPM Events*”.
- Upon receiving a SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg’ defined in Appendix H included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP INVITE request as described in section 8.2.1.2 “*Handle a Large Message Mode*”. If any additional strings (e.g. IARIs) are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.
- Upon receiving a SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ defined in Appendix H included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP INVITE request as described in section 8.2.2.1 “*Handle a CPM Session Invitation*”. If any additional strings (e.g. IARIs) are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.
- Upon receiving a SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer’ defined in Appendix H included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP INVITE request as described in section 8.2.3.1 “*Handle a CPM File Transfer*”. If any additional strings (e.g. IARIs) are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.
- Upon receiving a SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg’ defined in Appendix H included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP INVITE request as:
    - described in section 6.7.3.2 “*Receiving One Time CPM Events*”, when the SDP directional media attribute is set to a=sendonly, or
    - described in section 6.7.3.4 “*Receiving a Bi-directional Session for CPM Events Invitation*”, when the SDP directional media attribute is set to a=sendrecv.

8.2.1 CPM Standalone Message Handling

8.2.1.1 Handle a Pager Mode CPM Standalone Message and SIP IMDNs

Upon receiving a SIP MESSAGE request with one of the CPM Feature Tag listed in section 8.2 "*Procedures in the Originating Network*" included in the Accept-Contact header.

1. The CPM Participating Function SHALL check whether the authenticated originator's CPM Address is allowed to send the request and if not, the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "127 Service not authorised" in the response according to the rules and procedures of [RFC3261];

   Otherwise, continue with rest of the steps;

2. If the CPM Participating Function requires a specific User Agent version to be supported, the CPM Participating Function SHALL check the "User Agent" header field to determine if the CPM Participating Function supports the User Agent version and if not, the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "132 Version no supported" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with rest of the steps;

3. If the CPM Participating Function does not allow anonymity and anonymity is requested, the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text according set to "119 Anonymity not allowed" in the response according to the rules and procedures of [RFC3261]

   Otherwise, continue with the rest of the steps;

4. If the "Expires" header is included, the CPM Participating Function SHALL check if the message is still valid. If the message is no longer valid, the message is handled as specified in [RFC3428].

   Otherwise continue with the rest of the steps;

5. The CPM Participating Function SHALL check if there are external contents in the MIME body of the message, for each Content-Type header set to the value "message/external-body" and whose access-type is set to the value "URL" and whose URL contains the parameter "action=fetch", the CPM Participating Function

   a. SHALL fetch the Media Object, CPM Standalone Message, CPM File Transfer History, CPM Session History or CPM Conversation History indicated by the URL from the Message Storage Server as described in [OMA-CPM_TS_MessageStorage].

   b. SHALL replace the value "message/external-body" in the Content-Type of CPIM body header with an appropriate value depending on the fetched data type.

      NOTE: The value in Content-Type header of CPIM body can be obtained from the stored data's Content-Type value.(e.g. if stored data is a text file whose Content-Type is "text/plain", the "text/plain" can be copied to the Content-Type header in CPM Standalone Message.)

If the Content-Type header of CPIM body is a multipart (e.g. multipart/related, multipart/mixed), the CPM Participating Function SHALL add the CPIM cpm.Payload-Type header set to the list of content types found in the body, in the same order in which they appear.

c. SHALL include the fetched data in the message/cpim body.

6. The CPM Participating Function SHALL check in the originator's user preferences retrieved from XDMS as described in 8.4.1 "*Retrieving User Preferences*" if there is a rule in [OMA-XDM-Policy] in which the "enabler" attribute of the '<service>' sub-element inside the '<service-list>' element of the conditions part of the rule is set to "CPM" and the '<action>' element of the rule contains an '<allow-offline-storage>' sub-element set to "true". If such a rule exists , it SHALL execute the processing described in 8.5 "*Record CPM Conversation History*";
7. The CPM Participating Function:
   i. SHALL check if:
      1. the P-Preferred-Service header field is present and carries the value of one of the CPM Feature Tags such as 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg', or 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.group', or 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg', or 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group', or'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session', or 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group', or 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer', or 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer.group'. In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add that value in the P-Asserted-Service header field; or
      2. if P-Asserted-Service header field is present and does not contain a value of the CPM Feature Tag of 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg', or 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.group', or 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg', or 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group', or'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session', or'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group', or 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer', or 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer.group', it MAY respond with a 403 'Forbidden' SIP response;
8. The CPM Participating Function SHALL check the message size.
   A) If the maximum size of the message (after insertion of external content in step 5 is not larger than 1300 bytes, the CPM Participating Function:
      i. If the destination address is neither SIP URI address nor TEL URI address, and the request does not include a list of recipients for an Ad-hoc Group:
         1. the CPM Participating Function SHALL forward the SIP MESSAGE directly to the ISF, or via the SIP/IP Core as described in section 6.5 "*Communicating with the ISF and IWF".*
      ii. Otherwise, the CPM Participating Function SHALL forward the SIP MESSAGE request to the SIP/IP core.
      iii. Upon receiving an error response to the SIP MESSAGE request that is one of
         - SIP 404 "Not Found",
         - 405 "Method Not Allowed"
         - 410 "Gone"
         - 414 "Request URI Too Long"
         - 415 "Unsupported Media Type"
         - 416 "Unsupported URI Scheme"
         - SIP 488 "Not Acceptable Here" and
         - SIP 606 "Not Acceptable",

the CPM Participating Function:

1. SHALL check if service provider policy allows to initiate interworking;
2. If interworking is allowed, it SHALL send the SIP MESSAGE request directly to the ISF, or via the SIP/IP Core, as described in section 6.5 "*Communicating with the ISF and IWF"*.

Otherwise, continue with the rest of the steps.

iv. Upon receiving a SIP final response, the CPM Participating Function SHALL use this SIP final response as the final response for the delivery of the CPM Standalone Message.

B) Otherwise, when the message size is larger than 1300 bytes the CPM Participating Function:

a) SHALL generate an initial SIP INVITE request according to the rules and procedures of [RFC3261], with the following clarifications:

i. SHALL copy the Accept-Contact header included in the incoming SIP MESSAGE request to the outgoing SIP INVITE request, and replace the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg' value in the g.3gpp.icsi-ref media feature tag by 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg' percent encoded as per [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI"*. If an Accept-Contact header field containing a +sip.instance is present, it SHALL NOT copy it in the generated SIP INVITE;

ii. SHALL check if:

1. the P-Preferred-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg'. In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add P-Asserted-Service header field set to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg'; or
2. the P-Preferred-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.group'. In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add P-Asserted-Service header field set to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group'; or
3. if P-Asserted-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg', it SHALL replace the value in P-Asserted-Service with 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg'; or
4. if P-Asserted-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.group', it SHALL replace the value in P-Asserted-Service with 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group'

iii. SHALL copy the authenticated originator's CPM Address to the outgoing SIP INVITE request if it was included in the incoming SIP MESSAGE request;

iv. SHALL copy the Reply-To header to the outgoing SIP INVITE request if it was included in the incoming SIP MESSAGE request.

v. SHALL include a User-Agent header to indicate the CPM release version as specified in Appendix D "*Release Version in User-agent and Server headers*";

vi. SHALL copy the Privacy header to the outgoing SIP INVITE request if it was included in the incoming SIP MESSAGE request;

vii. SHALL include the option tag 'timer' in the Supported header;

viii. SHALL include the "Message-Expires" header in the outgoing SIP INVITE request as defined in Appendix C "*CPM-defined Header fields*" using the value from the incoming Expires header if it was included in the incoming SIP MESSAGE request.

ix. SHALL include the Session-Expires header with the refresher parameter set to "uac" according to the rules and procedures of [RFC4028];

x. SHALL set the delta-seconds value to a value lower than 1800 in the Session-Expires header according to the rules and procedures of [RFC4028], based on a service provider configured value for a Large Message Mode CPM Standalone Message SIP session;

xi. SHOULD NOT include the Min-SE header field according to the rules and procedures of [RFC4028];

xii. SHALL copy the Request-URI to the outgoing SIP INVITE request from the incoming SIP MESSAGE request;

xiii. SHALL copy the MIME resource-list body to the outgoing SIP INVITE request if it was included in the incoming SIP MESSAGE request;

xiv. SHALL copy the Conversation-ID to the outgoing SIP INVITE request from the incoming SIP MESSAGE request;

xv. SHALL copy the Contribution-ID to the outgoing SIP INVITE request from the incoming SIP MESSAGE request;

xvi. SHALL, if present, copy the InReplyTo-Contribution-ID to the outgoing SIP INVITE request from the incoming SIP MESSAGE request;

xvii. SHALL include in the SIP INVITE request a MIME SDP body as an SDP offer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714] with the following clarifications:

1. SHALL set the SDP directional media attribute to a=sendonly;
2. SHALL set the content type as a=accept-types:message/cpim;
3. SHALL set MSRP URI for the MSRP connection setup as a=path:MSRP URI;
4. SHALL set the size as a=file-selector:size:actual message size;
5. SHALL set the a=setup attribute as “actpass”;

b) If the destination address is neither SIP URI address nor TEL URI address, the CPM Participating Function SHALL send the SIP INVITE request directly to the ISF as described in section 6.5 “*Communicating with the ISF and IWF”*.

c) Otherwise, the CPM Participating Function SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

d) Upon receiving a SIP 200 “OK” response to the SIP INVITE request, the CPM Participating Function:

i. SHALL start the SIP session timer using the value received in the Session-Expires header according to the rules and procedures of [RFC4028];

ii. SHALL act as MSRP client according to [RFC6135];

iii. If it has received in the SDP answer of the 200 “OK” response an a=setup attribute:

1. with a value of “active”, then it SHALL act as an “passive” endpoint and wait for the other endpoint to initiate the transport connection according to [RFC6135];
2. Otherwise, it SHALL act as an "active" endpoint to open the transport connection according to [RFC6135] ;

iv. SHALL establish the MSRP connection according to the MSRP connection parameters in the SDP answer received in the SIP 200 “OK” response according to [RFC6135];

v. SHALL generate and send a SIP ACK request as an acknowledgement of the final response according to the rules and procedures of [RFC3261];

vi. SHALL generate one or more MSRP SEND requests (depending on whether chunking is required) according to the rules and procedure of [RFC4975] taking into account the maximum chunk size negotiated according to section 5.2.1, if any, with the following clarification:

1. SHALL set To-Path header according to the MSRP URI(s) received in the answer SDP;
2. SHALL set the content type as Content-Type = message/cpim and SHALL copy the second Content-Type header field. If the CPIM wrapped message body includes a multipart MIME (e.g. multipart/related, multipart/mixed), then:
   i. SHALL add a CPIM cpm.Payload-Type header field, set to the list of Content-Types included in the payload of the message, in the same order in which they appear in the body;
3. If the original SIP MESSAGE request requested for a disposition notification, it SHALL set the disposition notification request in the MSRP SEND request as an extension header as described in section 9.2.15 “*Disposition Notification*”;
4. SHALL send the MSRP SEND request(s) on the established MSRP connection.

vii. When the last MSRP SEND request (representing the last chunk) has been sent and acknowledged, the CPM Participating Function:

a. SHALL generate a SIP BYE request according to the rules and procedures of [RFC3261];

b. SHALL send the SIP BYE request according to the rules and procedures of SIP/IP core;

c. SHALL use the status of the MSRP send request(s) as the final response for the delivery of the CPM Standalone Message.

d. Upon receiving a SIP response to the SIP BYE:

a. If the response is an error response from the SIP/IP core, e.g. SIP 404 “Not Found”, SIP 488 “Not Acceptable Here” or SIP 606 “Not Acceptable”, or upon receiving an error response to an MSRP SEND request sent in step c, the CPM Participating Function:

i. SHALL, if interworking has not yet been attempted, and if allowed according to service provider policy, send the SIP INVITE request directly to the ISF as described in section 6.5 “*Communicating with the ISF and IWF”*;

ii. Otherwise, the CPM Participating Function SHALL send the appropriate SIP error response for the received SIP MESSAGE request back to the served CPM Client, in accordance with [RFC3428].

9. If the SIP final response, to the SIP MESSAGE request sent in step A) above or to the SIP BYE request sent in step B) above, is a SIP “200” OK response, the CPM Participating Function checks:

i. If the user preference for recording, or service provider policies, were set to record the CPM Conversation History, then the CPM Participating Function:

a. SHALL store the CPM Standalone Message into the Message Storage Server according to procedures specified in section 6.3.1 “*Object Store Operation*” of [OMA-CPM_TS_MessageStorage] and retrieve an UID from the Message Storage Server;

b. If recording was successful, SHALL include in the SIP final response the Message-UID header set to the retrieved UID information as specified in Appendix C.1.6 “*Message-UID*”;

ii. SHALL send a SIP 200 “OK” final response for the received SIP MESSAGE request or SIP BYE request back to the served CPM Client, in accordance with [RFC3428];

iii. The MSRP media handling by the originating CPM Participating Function with the CPM Client of the served CPM User SHALL be done as described in section 8.6 “*Media Plane Handling*”;

iv. Upon successfully sending the first MSRP message of the Large Mode CPM Standalone Message (i.e. a MSRP 200 “OK” response was received on the terminating leg), the CPM Participating Function SHALL check the service provider policy to determine if it needs to apply the direct delivery procedure for the originated CPM Message. To perform the direct delivery of the originated CPM Message, the CPM Participating Function SHALL follow the procedures section 8.2.1.2 “*Handle a Large Message CPM Standalone Message*” starting with step A.

If the CPM Message was delivered via SIP MESSAGE request to the terminating network and a SIP 200 “OK” response was received from the terminating leg, the CPM Participating Function SHALL check the service provider policy to determine if it needs to continue with the direct delivery procedure for the originated CPM Message. To perform the direct delivery, the CPM Participating Function SHALL send that originated CPM Message towards all suitable registered CPM Clients of the originating CPM User, as follows:

1) If the CPM Participating Function requires a specific User-Agent version, it SHALL determine if any of the suitable CPM User’s registered Clients supports that version, besides the CPM Client that originated the message;

2) If no suitable CPM User’s registered Clients support the specific User-Agent version required by the CPM Participation Function, then it SHALL skip the rest of the direct delivery procedure.
Otherwise it SHALL continue with the remaining steps;

3) SHALL generate a SIP MESSAGE request as follows:

i. SHALL copy the SIP header fields received in the original SIP MESSAGE request;

ii. SHALL add the CPIM Message-Direction header field set to value “sent” in the CPIM wrapper of the payload;

iii. SHALL copy the original CPIM header fields of the sent CPM Message and the payload;

iv. SHALL send out the SIP MESSAGE request in one of the following two ways:

a. to the SIP/IP Core for further forking to all registered CPM User's Clients, or;

b. to each of the suitable CPM User's registered CPM Clients except for the CPM Client that originated the CPM Message. For each suitable registered CPM User's Client:

i. it SHALL set the Request-URI to the suitable registered CPM Client's identity, which is either its public GRUU (if supported) or its CPM User's Address;

ii. If public GRUU is not supported, then it SHALL also include a new Accept-Contact header carrying only the +sip.instance parameter and the instance identifier value of the suitable CPM User's registered Client, as well as the tags "explicit" and "require" as described in [RFC3841];

4) SHALL send the SIP MESSAGE request(s) according to the rules and procedures of the SIP/IP core.

8.2.1.2 Handle a Large Message Mode CPM Standalone Message

Upon receiving an initial SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg' included in the Accept-Contact header corresponding to Large Message Mode CPM Standalone Message, the CPM Participating Function:

1. SHALL check whether the authenticated originator's CPM Address is of a CPM User that is allowed to send the request and if not, the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "127 Service not authorised" in the response according to the rules and procedures of [RFC3261];

   Otherwise, continue with rest of the steps;

2. If the CPM Participating Function requires a specific User Agent version to be supported, the CPM Participating Function SHALL check the "User Agent" header field to determine if the CPM Participating Function supports the User Agent version and if not, the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "132 Version not supported" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with rest of the steps;

3. If the CPM Participating Function does not allow anonymity and anonymity is requested, the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "119 Anonymity not allowed" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;

4. If the "Message-Expires" header as defined in Appendix C "*CPM-defined Header fields*" is included, the CPM Participating Function SHALL check if the message is still valid. If the message is no longer valid, the message is handled in the same way as for an "Expires" header in a SIP MESSAGE as defined in [RFC3428];

   Otherwise continue with the rest of the steps;

5. If the "Session-Expires" header field is included with the refresher parameter set to "uac",the CPM Participating Function SHALL handle the header field according to procedures of [RFC4028]. If the"Session-Expires" header field with the refresher parameter set to "uas", the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "122 Function not allowed";

6. SHALL behave as a B2BUA according to the rules and procedures of [RFC3261] for the duration of the SIP session;

7. SHALL generate a SIP INVITE request according to the rules and procedures of [RFC3261] with the following clarifications:

   a. The CPM Participating Function SHALL include the Request-URI received in the SIP INVITE request;

   b. The CPM Participating Function SHALL copy the Accept-Contact header included in the incoming SIP INVITE request to the outgoing SIP INVITE request percent encoded as per [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI"*;

   c. The CPM Participating Function SHALL check if in the received SIP INVITE:

      i. the P-Preferred-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg'. In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add P-Asserted-Service header field and set it to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg'; or

ii. the P-Preferred-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group'. In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add P-Asserted-Service header field and set it to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group'; or

iii. if P-Asserted-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg' or the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group'. If another value is found, the CPM Participating Function MAY reject the SIP INVITE with a SIP 403 "Forbidden" response,

d. The CPM Participating Function SHALL copy the Contact header of the incoming SIP INVITE request to the outgoing SIP INVITE request;

e. SHALL include a Session-Expires header field with the refresher parameter set to "uac";

f. SHALL set the delta-seconds value to a value lower than 1800 in the Session-Expires header according to the rules and procedures of [RFC4028], based on a service provider configured value for a Large Message Mode Standalone Message SIP session;

g. SHOULD NOT include the Min-SE header field according to the rules and procedures of [RFC4028];

h. The CPM Participating Function SHALL insert a URI identifying its own address in the Contact header;

i. The CPM Participating Function SHALL include a MIME SDP body received in the SIP INVITE request as an SDP offer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714] with the following clarification. The CPM Participating Function:

   i. SHALL include a media line proposing MSRP media parameters;

   ii. SHALL populate its own MSRP URI as a=path:MSRP URI;

   iii. SHALL set the SDP directional media attribute to a=sendonly;

   iv. SHALL set the size as a=file-selector:size:actual message size;

   v. SHALL set the a=setup attribute as "actpass";

   vi. If not configured for MSRP interoperability, or if no MSRP interoperability is needed, it SHALL include an SDP 'msrp-cema' attribute in the MSRP media description of the SDP. Otherwise, it SHALL proceed according to section 5.2.1.4 "*Handling of Media connection parameters for MSRP".*

8. SHALL, if the destination address is neither a SIP URI address nor a TEL URI address, send the SIP INVITE request directly to the ISF as described in section 6.5 "*Communicating with the ISF and IWF"*. Otherwise, the CPM Participating Function SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

Upon receiving a SIP 200 "OK" response to the SIP INVITE request sent in step 8 above, the CPM Participating Function:

1. SHALL generate a SIP 200 "OK" response according to the rules and procedures of [RFC3261];
2. SHALL copy the Contact header field of the received SIP 200 "OK" response, if the 'isfocus' feature parameter as described in Appendix C.1.7 "Contact header field values" is present, otherwise SHALL include a Contact header field with a URI identifying its own address;
3. SHALL include a Server header to indicate the CPM release version as specified in Appendix D "*Release Version in User-agent and Server headers*";
4. SHALL include a Session-Expires header field based on the values received in the SIP 200 "OK", according to the rules and procedures of [RFC4028];
5. SHALL include the SDP received in the response to the SIP INVITE request as answer SDP according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714] with the following clarifications:
   a. The CPM Participating Function SHALL include a media line proposing MSRP media parameters;
   b. The CPM Participating Function SHALL populate its own MSRP URI as a=path:MSRP URI;
   c. The CPM Participating Function SHALL set the SDP directional media attribute to a=recvonly;
   d. The CPM Participating Function SHALL set the a=setup attribute as "passive";
   e. If the SDP 'msrp-cema' attribute in the MSRP media description was present in the original SIP INVITE request from the CPM Client, the CPM Participating Function SHALL include the SDP 'msrp-cema' attribute in the SDP answer and otherwise SHALL omit it and SHALL proceed according to section 5.2.1.4 "*Handling of Media connection parameters for MSRP"*.

6. SHALL send the SIP 200 “OK” response according to the rules and procedures of SIP/IP core and SHALL start listening for incoming MSRP media;
7. SHALL check in the originator’s user preferences retrieved from XDMS as described in 8.4.1 “*Retrieving User Preferences*” if there is a rule in [OMA-XDM-Policy] in which the “enabler” attribute of the ‘<service>’ sub-element inside the ‘<service-list>’ element of the conditions part of the rule is set to “CPM” and the ‘<action>’ element of the rule contains an ‘<allow-offline-storage>’ sub-element set to “true”. If such a rule exists, it SHALL execute the processing described in 8.5 “*Record CPM Conversation History*”.

Upon receiving an error response from the SIP/IP core that is one of:

- SIP 404 “Not Found” or
- 405 “Method Not Allowed”
- 410 “Gone”
- 414 “Request URI Too Long”
- 415 “Unsupported Media Type”
- 416 “Unsupported URI Scheme”
- SIP 488 “Not Acceptable Here” and
- SIP 606 “Not Acceptable”,

the CPM Participating Function SHALL, if allowed according to service provider policy, send the SIP INVITE request generated in step 7 above directly to the ISF as described section 6.5 “*Communicating with the ISF and IWF”* if interworking is enabled per service provider policies and/or by user preferences. Upon receiving a SIP 200 “OK” response from the ISF, the same handling applies as when receiving a SIP 200 “OK” response from the SIP/IP Core to the SIP INVITE request sent in step 8 above. When receiving a non-2XX final response from the ISF, the CPM Participating Function SHALL return the equivalent response to the client.

Upon receiving a SIP CANCEL request, the CPM Participating Function:

1. SHALL act as UAS to handle the SIP CANCEL request according to the rules and procedures of [RFC3261]; and,
2. SHALL act as UAC to cancel the sent SIP INVITE request according to the rules and procedures of [RFC3261].

Upon receiving a SIP ACK request, the CPM Participating Function SHALL forward the SIP ACK request to SIP/IP core according to the rules and procedures of the SIP/IP core.

Upon successfully sending the first MSRP message of the Large Mode CPM Standalone Message (i.e. a MSRP 200 “OK” response was received on the terminating leg), the CPM Participating Function SHALL check the service provider policy to determine if it needs to continue with the direct delivery procedure for the originated CPM Message. To perform the direct delivery of the originated CPM Message, the CPM Participating Function SHALL send the originated Large Message CPM Standalone Message towards all suitable registered CPM Clients of the originating CPM User, as follows:

A. If the CPM Participating Function requires a specific User-Agent version, it SHALL determine if any of the suitable CPM User’s registered Clients supports that version, besides the CPM Client that originated the message;

B. If no suitable CPM User’s registered Clients are found (e.g., none supports the specific User-Agent version required by the CPM Participation Function), then it SHALL skip the rest of the direct delivery procedure.
Otherwise it SHALL continue with the remaining steps;

C. SHALL proceed with the direct delivery procedure towards each of the remaining suitable registered CPM Clients of the CPM User. It SHALL generate a SIP INVITE request to each suitable registered CPM Client as specified above starting from step 5 of receiving a SIP INVITE, with the following clarifications:

    1) SHALL copy the SIP header fields received in the original SIP INVITE request;
    2) SHALL insert a URI identifying its own address (CPM Participating Function) in the Contact header field;
    3) SHALL send out a SIP INVITE request to each of the suitable CPM User’s registered CPM Clients except for the CPM Client that originated the CPM Message. For each SIP INVITE request towards a suitable registered CPM User’s Client:

        i. it SHALL set the Request-URI to the suitable registered CPM Client’s identity, which is either its public GRUU (if supported) or its CPM User’s Address;
        ii. If public GRUU is not supported, then it SHALL also include a new Accept-Contact header carrying only the +sip.instance parameter and the instance identifier value of the suitable CPM User’s registered Client, as well as the tags “explicit” and ”require” as described in [RFC3841];

D. SHALL send the SIP INVITE request(s) according to the rules and procedures of the SIP/IP core;

E. When a SIP response is received from a suitable registered CPM Client of the served originating CPM User, the CPM Participating Function:

   a. If a SIP 200 “OK” response is received, it SHALL return a SIP ACK and then it SHALL proceed with setting up the MSRP session;

   b. If an error SIP response is received, then it SHALL not continue with the remaining steps;

   c. Once the MSRP session is setup, SHALL add the CPIM Message-Direction header field set to value “sent” in the CPIM wrapper of each MSRP SEND payload;

   d. SHALL copy the original CPIM header fields of the sent CPM Message, including the CPIM cpm.Payload-Type header field, if present.

The MSRP media handling by the originating CPM Participating Function with any one of the CPM Clients of the served CPM User SHALL be done as described in section 8.6 “*Media Plane Handling*”.

Upon receiving a SIP BYE request from either the originating CPM User or the terminating network, the CPM Participating Function:

1. If MSRP message delivery to the terminating side was accepted successfully (e.g., a MSRP 200 “OK” response was received for the all chunks belonging to a MSRP message) and if the user preference was set to record the CPM Conversation History, SHALL store the CPM Standalone Message to the Message Storage Server according to procedures specified in section 6.3.1 “*Object Store Operation*” of [OMA-CPM_TS_MessageStorage] and retrieve an UID from the Message Storage Server;
2. SHALL respond to the SIP BYE request as described in [RFC3261] and if the message was recorded, SHALL include in the 200 “OK” response the Message-UID header set to the retrieved UID information as specified in Appendix C.1.6 “*Message-UID*”;
3. SHALL forward the SIP BYE request to the SIP/IP core according to rules and procedures of the SIP/IP core.

8.2.2 CPM Session Handling

A CPM Participating Function in the originating network may or may not stay in the media path depending on service provider’s policies. The policies may be static such that the CPM Participating Function always stays or always does not stay in the media path, or the policies may depend on parameters such as:

1. If a CPM User has a subscription to the Message Storage Server and a user preference is set to record CPM Conversation History when he sends a CPM Session Invitation;
2. If the CPM Participating Function needs to do charging.
3. To support deferred delivery if enabled.

When a CPM Participating Function stays in the media path acting as a back-to-back user agent (B2BUA), the following applies:

a) The CPM Participating Function acts as a user agent for both SIP sessions it connects. The B2BUA behaves as a user agent server for the incoming side and as a user agent client for the outgoing side(s).

b) The CPM Participating Function correlates both sides of the SIP session, by mapping the corresponding SIP session identities of the SIP sessions, and it maintains session state for all SIP sessions it handles, thus enabling end-to-end communication.

c) The CPM Participating Function, based on service providers policies, MAY process CPM events.

8.2.2.1 Handle a CPM Session Invitation

Upon receiving a SIP INVITE request for a CPM 1-1 Session or a CPM Group Session:

1. The CPM Participating Function SHALL check whether the authenticated originator’s CPM Address is of a CPM User that is allowed to send the request and if not, the CPM Participating Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “127 Service not authorised” in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with rest of the steps;
2. If the CPM Participating Function requires a specific User Agent version to be supported, the CPM Participating Function SHALL check the “User Agent” header field to determine if the CPM Participating Function supports the User Agent version as defined in Appendix D “*Release Version in User-agent and Server headers*” and if not, the

CPM Participating Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “132 Version not supported” in the response according to the rules and procedures of [RFC3261].

Otherwise, continue with rest of the steps;

3. If the CPM Participating Function does not allow anonymity and anonymity is requested, the CPM Participating Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “119 Anonymity not allowed” in the response according to the rules and procedures of [RFC3261]

   Otherwise, continue with the rest of the steps;

4. If the CPM Participating Function stays in the media path, the CPM Participating Function SHALL check in the received SIP INVITE:

   a. if the P-Asserted-Service header field is present:

      ii. and carries the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ or the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group’, then it SHALL continue with the next step (4b).

      iii. otherwise, the CPM Participating Function MAY reject the SIP INVITE with a SIP 403 “Forbidden” response and SHALL not continue with the next steps;

   b. SHALL generate a SIP INVITE request according to the rules and procedures of [RFC3261];

   c. If the P-Preferred-Service header field is present and:

      ii. carries the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ remove the P-Preferred-Service header field and add P-Asserted-Service header field and set it to ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ to the generated SIP INVITE; or

      iii. carries the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group’ remove the P-Preferred-Service header field and add P-Asserted-Service header field and set it to ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group’ to the generated SIP INVITE;

      iv. otherwise, if the P-Preferred-Service header field contains another value, the CPM Participating Function SHALL remove the P-Preferred-Service header field and,

         1. add a P-Asserted-Service header based on the other information in the SIP INVITE request (i.e. set it to the CPM Feature Tag in the Accept-Contact header field); or

         2. add a P-Asserted-Service header based on service provider configuration;

   d. SHALL copy from the original SIP INVITE: the received Request-URI, the header fields Conversation-ID, Contribution-ID, InReplyTo-Contribution-ID and Subject and also the received recipient-list from the body of the original SIP INVITE;

   e. SHALL behave as a B2BUA according to the rules and procedures of [RFC3261] for the duration of the CPM Session;

   f. SHALL generate a SIP INVITE request according to the rules and procedures of [RFC3261];

   g. SHALL include a User-Agent header to indicate the OMA CPM release version of the CPM Participating Function as specified in Appendix D “*Release Version in User-agent and Server headers*”;

   h. SHALL copy the Contact header and the Accept-Contact header of the incoming SIP INVITE request to the outgoing SIP INVITE request;

   i. SHALL set the URI provided in the Contact header to provide a URI as per [3GPP TS 24.229].

   j. SHALL add a Supported header field with the option tag ‘timer’ and if included in the original SIP INVITE request also a ‘recipient-list-invite’ tag;

   k. SHALL include a Session-Expires header field with the refresher parameter set to “uac”;

   l. SHALL set the delta-seconds value to a value of 1800 or lower in the Session-Expires header according to a service provider configured value and to the rules and procedures of [RFC4028];

   m. SHALL include an SDP body as an SDP offer in the SIP INVITE request based on the received SDP from the originating client, as described in section 5.2.1.2 “*SDP Handling at Intermediate Nodes*”;

   n. If the destination address is neither SIP URI address nor TEL URI address, the CPM Participating Function SHALL send the SIP INVITE request directly to the ISF as described in section 6.5 “*Communicating with* *the ISF and IWF"*. Otherwise the CPM Participating Function SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

5. If the CPM Participating Function does not stay in the media path, the CPM Participating Function:
    a. SHALL act as a SIP proxy according to the rules and procedures of [RFC3261] for the duration of the CPM Session;
    b. SHALL include a Record-Route header containing a URI identifying its own address; and,
    c. If the destination address is neither SIP URI address nor TEL URI address, the CPM Participating Function SHALL forward the SIP INVITE request directly to the ISF as described in section 6.5 "*Communicating with the ISF and IWF"*. Otherwise, the CPM Participating Function SHALL forward the SIP INVITE request according to the rules and procedures of [RFC3261] and the SIP/IP core.

Upon receiving a SIP 200 "OK" response from the terminating leg, if the CPM Participating Function is acting as a B2BUA, the CPM Participating Function:

1. SHALL, when a SIP 200 "OK response was received from the ISF, check whether the all offered Media Streams have been accepted. If not, the CPM Participating Function MAY attempt to establish a new session for the rejected Media Streams. To establish the new session the CPM Participating Function:
    a. SHALL check if service provider policy allows to establish a new session for the new Media Stream;
    b. If allowed, SHALL generate a SIP INVITE request with the following details;
        i. SHALL perform the above step 4 with the exception of step 4-f and step 4-g;
        ii. SHALL include SDP parameters about the rejected Media Streams in the SDP body, as described in section 5.2.1.2 "*SDP Handling at Intermediate Nodes*". The CPM Participating Function SHALL set the a=setup attribute to the value of "actpass" according to [RFC6135];
        iii. SHALL send the SIP INVITE request directly to the ISF.

        Otherwise, continue with the rest of steps;
2. SHALL generate a SIP 200 "OK" response towards the originating CPM Client;
3. SHALL include a Server header to indicate the OMA CPM release version of the CPM Participating Function as specified in Appendix D "*Release Version in User-agent and Server headers*";
4. SHALL include the received SDP body as an SDP answer based on the SDP answer received in the SIP 200 "OK" response as described in section 5.2.1.2 "*SDP Handling at Intermediate Nodes".* The CPM Participating Function SHALL set the a=setup attribute to the value of "passive" according to [RFC6135];
5. SHALL copy the Contact header field of the received SIP 200 "OK" response field, if the 'isfocus' feature parameter as described in Appendix C.1.7 "Contact header field values" is present, otherwise SHALL include Contact header field with a SIP URI that can resolve back to the original SIP URI in the received Contact header;
6. SHALL handle the Media Plane parameters as described in section 8.6 "*Media Plane Handling*";
7. SHALL send the SIP 200 "OK" response to the CPM Client according to the rules and procedures of SIP/IP core; and wait for the SIP ACK request from the CPM Client and SHALL propagate it towards the leg on the terminating side;
8. SHALL check the <actions> element <allow-offline-storage> retrieved from XDMS as described in 8.4.1 "*Retrieving User Preferences*", and if set to "true", or if the recording is enabled, it SHALL execute the processing described in 8.5 "*Record CPM Conversation History*", i.e. the storage of the SIP INVITE request data into the Session info Object. For the CPM Group Session, it also stores the Group State Object containing the CPM Group Session Identity and the Participant list as received in the body of the SIP INVITE request;
9. If service provider policy supports the direct delivery of originated chat messages to all suitable CPM User's registered Clients, then the CPM Participating Function:
    a. SHALL, if the CPM Participating Function requires a specific User-Agent version, determine if any of the suitable CPM User's registered Clients, excluding the one that originated this CPM Session, supports that version.
        i. If no suitable CPM User's registered Clients support the specific User-Agent version required by the CPM Participation Function, then it SHALL skip the rest of this procedure.
    b. SHALL generate a SIP INVITE to each of the other suitable CPM User's registered Clients, as follows:
        i. SHALL set the recipient CPM User's address with the device identifier of the targeted CPM Client;

        ii. SHALL copy the rest of the SIP header fields received in the original SIP INVITE request;

    c. SHALL send each SIP INVITE request according to the rules and procedures of the SIP/IP core;

    d. SHALL follow the CPM Participating Function terminating side procedures as described in section 8.3.2.1 "*Handle a CPM Session Invitation*" to handle the SIP responses and to establish the MSRP sessions to all other suitable CPM User's registered Clients that will accept the SIP session.

Upon receiving a SIP 200 "OK" response, when the CPM Participating Function is acting as a SIP proxy, the CPM Participating Function:

1. SHALL forward the SIP 200 "OK" response toward the initiating CPM Client according to the rules and procedures of [RFC3261] and the SIP/IP core; and,
2. SHALL continue to act as a SIP proxy for the duration of the CPM Session.

Upon receiving an error response that is not from one of the other CPM User's suitable clients, that is one of

- SIP 404"Not Found",
- 405 "Method Not Allowed"
- 410 "Gone"
- 414 "Request URI Too Long"
- 415 "Unsupported Media Type"
- 416 "Unsupported URI Scheme"
- SIP 488 "Not Acceptable Here" or
- SIP 606 "Not Acceptable",

based on service provider policies, the CPM Participating Function:

1. SHALL check if service provider policy allows to initiate interworking on the originating side;
2. If interworking is allowed, it SHALL send the SIP INVITE request directly to the ISF as described in section 6.5 "*Communicating with the ISF and IWF"*whereby the CPM Participating Function SHALL handle a SIP 200 "OK" response from the ISF in the same way as a SIP 200 "OK" response received from the terminating leg and forward other SIP final responses to the originating CPM Client;
3. Otherwise, it SHALL forward the SIP INVITE error response towards the originating CPM Client.

Upon receiving a SIP ACK request, the CPM Participating Function SHALL forward the SIP ACK request to the SIP/IP core according to the rules and procedures of the SIP/IP core and if the ACK was received on the terminating leg, the CPM Particpating Function SHALL establish the media plane towards the terminating entity.The MSRP media handling by the originating CPM Participating Function SHALL be done as described in section 8.6 "*Media Plane Handling*".

8.2.2.2 Handle a Cancel Request

Upon receiving a SIP CANCEL request, the CPM Participating Function performing as a B2BUA:

1. SHALL act as UAS according to the rules and procedures of [RFC3261];
2. SHALL cancel the SIP INVITE request towards the SIP/IP core according to the rules and procedures of [RFC3261].

When acting as a SIP proxy, rules and procedures as specified in [RFC3261] SHALL be applied.

8.2.2.3 Handle a SIP BYE Request

Upon receiving a SIP BYE request for an existing CPM Session from the originating leg with a served CPM Client:

1) a CPM Participating Function that acts as a SIP proxy:

    a. SHALL forward the received SIP BYE request along the Signalling Path and upon receiving the SIP 200 "OK" response associated with the SIP BYE request, the CPM Participating Function SHALL forward the SIP 200 "OK" response along the same Signalling Path as the SIP BYE request according to the rules and procedures of [3GPP TS24.229].

2) If the CPM Participating Function acts as a B2BUA then:

    a. SHALL check if a Reason header field is present in the SIP BYE requestIf a Reason header field is:

        A) If a Reason header field is not present (e.g. the served CPM Client is a CPM v1.0 Client, or SIP/IP Core network elements did not populate it), then the CPM Participating Function:

i. SHALLhandle it as a served CPM User initiated request, i.e. assuming a Reason header field with protocol=SIP;cause=200 and continue with step i. below.

B) If present, the CPM Participating Function SHALL check the values in the Reason header field to determine if the SIP BYE was generated by a served CPM Client, or another entity on its behalf, as follows:

i. If the Reason header field (as defined in [RFC3326]) has a protocol value set to SIP;cause=200; and MAY have a text parameter containing an explanatory String (e.g. "Call completed", or other), then the SIP BYE was generated by the CPM Client as result to CPM User's explicitly leaving the CPM Session, or by the CPM IWF entity acting on behalf of the CPM User that has left;

1) If there are other CPM User's suitable CPM Clients connected to this CPM Session (e.g. the service provider policy supports the direct delivery of originated chat messages to all suitable CPM User's registered Clients) then:

a) SHALL generate a SIP BYE request towards each of the connected suitable CPM Clients to terminate the CPM Session;

b) SHALL send the SIP BYE request according to the rules and procedures of [RFC3261] to the remaining CPM User's connected Clients;

c) Upon receiving the SIP 200 "OK" response to each of the SIP BYE requests sent, it SHALL release the Media Plane resources associated with the CPM Session toward the connected suitable CPM Client on the leg on which the SIP response was received.

2) If the affected CPM Session is being interworked, the CPM Participating Function:

a) SHALL generate SIP BYE requests for each of the associated IWFs according to the rules and procedures of [RFC3261];

b) SHALL send these SIP BYE requests to the IWFs according to the rules and procedures of [RFC3261].

3) if the SIP BYE request was received in a CPM Group Session where there is no active session with the CPM Controlling function, SHALL attempt to set up such a session. To achieve this, the CPM Participating Function:

a) SHALL generate a SIP INVITE request according to the rules and procedures of [RFC3261];

b) SHALL set the Request-URI to the address of the target CPM Group Session of CPM Long-lived Group Session;

c) SHALL include a User-Agent header to indicate the OMA CPM release version of the CPM Participating Function as specified in Appendix D "Release Version in User-agent and Server headers";

d) SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' 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;

e) SHALL include a Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' 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;

f) SHALL set the P-Asserted-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group';

g) SHALL include the CPM Address of the CPM User as the authenticated originator's CPM Address in Contact header field, as described in section 6.1 "*Authenticated Originator's CPM Address*";

h) SHALL add a Supported header field with the option tag 'timer';

i) SHALL include a Session-Expires header field with the refresher parameter set to "uac";

   j) SHALL include the Conversation-ID header field of the requested CPM Group Session;

   k) SHALL include the Contribution-ID header field of the requested CPM Group Session;

   l) SHALL set the delta-seconds value to a value of 1800 or lower in the Session-Expires header according to a service provider configured value and to the rules and procedures of [RFC4028];

   m) SHALL include an SDP body as an SDP offer in the SIP INVITE request based on the received SDP from the originating client, as described in section 5.2.1.2 "*SDP Handling at Intermediate Nodes*";

   n) SHALL send the generated SIP INVITE request towards the CPM Controlling Function;

   o) When receiving a SIP non-2XX final response to the SIP INVITE request, the CPM Participating Function SHALL send a SIP ACK request to confirm its reception according to [RFC3261] and go straight to step 6) a);

   p) When receiving a SIP 2XX Final Response, the CPM Participating Function SHALL send a SIP ACK request and continue with the rest of the steps (i.e. the Media Plane SHALL NOT be established);

4) SHALL generate a SIP BYE request towards the other CPM entity of the affected CPM Session (i.e. towards the terminating leg, to the CPM Controlling Function, or to the other CPM Participation Function) according to the rules and procedures of [RFC3261];

5) SHALL send the SIP BYE request according to the rules and procedures of [3GPP TS24.229].

6) Upon receiving a SIP 200 ”OK” response from the IWFs or the other CPM entity associated with the SIP BYE request, the CPM Participating Function:

   a) SHALL generate a SIP 200 "OK" response towards the other side, according to the rules and procedures of [RFC3261];

   b) If the action element <allow-offline-storage> is set to true and is using a CPM Session History Folder, the CPM Participating Function:

      i. MAY store based on service provider policy a Conference Information Object in the CPM Message Store Server under the CPM Group Session folder where it SHALL populate the latest list of known Participants and CPM Group Session Data in a conference-info body considering the rule defined in section 9.2.14.3 "*Sending Participant Information Notification*" for the use in a MSRP SEND request. The CPM Participating Function SHALL include in the <endpoint> element of the <user> element with the "yourown" attribute

         - the <status> element including the value "disconnected"
         - the <disconnection-info> element including
           - SHALL include <when> element including the date and time that the Participant left the CPM Session
           - SHALL include the <reason> element including the value "SIP;cause=200;text="Call completed"

      ii. MAY store based on service provider policy a create and store the Group State Object;

   c) SHALL send the SIP 200 "OK" response according to the rules and procedures of [3GPP TS24.229];

7) SHALL release all Media Plane resources associated with the CPM Session after sending the 200 "OK" response.

8) SHALL discard any remaining stored messages sent in the session to be for that CPM User;

9) SHALL for a CPM Group Session terminate any ongoing subscriptions for Participant Information that it may have with the CPM Controlling Function;

ii. If the Reason header field has a value of protocol:SIP;cause=503; (e.g. MAY have a text=”Service Unavailable”) then the SIP BYE request may have been generated by a node of the SIP/IP Core to indicate a connection loss with the served CPM Client, as defined in [3GPP TS24.229];

a. If there are other CPM User’s Clients connected into the CPM Session, then the error that occurred on the leg towards one of the CPM User’s Clients SHALL not affect the on-going CPM Session with the remaining connected clients. The CPM Participating Function:

   i. SHALL send a SIP 200 “OK” response towards the leg with the disconnected CPM User’s Client,

   ii. SHALL release the Media Plane resources associated with that leg,

   iii. and SHALL skip the remaining steps;

b. If no other suitable CPM User’s CPM Client(s) are connected, the CPM Participating Function:

   i. SHALL check relevant settings for the CPM User:

      1) if the action element includes an <allow-offline-storage> sub-element from user preferences is set to true for the CPM User, or

      2) if the action element includes an ‘<allow-store>’ sub-element set to “true”, or

      3) if the action element includes an ‘<allow-defer>’ sub-element set to “true”, or

      4) if the service provider policies allow deferral of the CPM Session, or

      5) if the service provider policies allow the CPM User to re-start CPM Long-lived Group Sessions.

   ii. If any one of them is true, then the CPM Participating Function SHALL remain in the media path of the active CPM Group Session, after a connection loss with the served CPM Client, and it:

      1) SHALL release the Media Plane resources towards the served CPM Client for this CPM session; and SHALL send a SIP SUBSCRIBE to the CPM Controlling Function of that CPM Group Session, as described in sect. 8.2.2.6 “*Handling of Participants Information*”. This will allow the CPM Participating Function to provide (and record, depending on the user preferences or service provider policies), the last list of participants to the CPM Group Session, and

      2) SHALL intercept all messages and disposition notifications exchanged throughout the duration of the CPM Group Session, for a later delivery to the CPM Client, and then for their recording in the CPM Message Storage of the CPM User.

      3) The CPM Participating Function SHALL remain in the CPM Group Session until it receives a SIP BYE from the CPM Controlling Function, or until the CPM Client re-joins and explicitly releases it as described in step I;

      4) SHALL terminate any ongoing subscriptions for Participant Information that it may have with the CPM Client according to the normal procedures in [RFC4575]. The leg towards the CPM Controlling Function SHALL either be maintained or be terminated in which case the CPM Participating Function SHALL start a new

subscription as defined in section 8.2.2.6 “Handling of Participants Information”.

iii. If the SIP BYE request carried a Reason header field with protocol=SIP and a different cause value than 200 or 503, and it was sent by one of the IWFs involved in the interworking of the CPM Session on behalf of the served CPM Client), the CPM Participating Function:

1. If the CPM Session is interworked towards multiple IWFs and the service provider policies indicate that only the Media Streams associated with the IWF that sent the SIP BYE request are to be closed, the CPM Participating Function SHALL send a SIP re-INVITE request for deleting the Media Streams associated with the IWF that sent the SIP BYE request to the CPM Client, according to the rules and procedures of [RFC3261] and the SIP/IP core.
2. Otherwise, if the CPM Session is interworked towards multiple IWFs and the service provider policies indicate closure of the entire CPM Session, or the CPM Session is interworked to a single IWF, the CPM Participating Function:
   a. If the action element <allow-offline-storage> is set to true, the CPM Participating Function:
      i. SHALL store the CPM Session to the Message Storage Server according to procedures specified in section 8.5.2 “*Record CPM Session*”;
   b. SHALL generate and send the SIP BYE request towards the terminating leg, according to the rules and procedures of [RFC3261].
   c. Upon receiving a SIP 200 ”OK” response from the CPM Client associated SHALL generate and send a SIP 200 “OK” response towards the IWF according to the rules and procedures of [RFC3261].
3. SHALL release all Media Plane resources associated with the CPM session, after receiving the 200 ”OK” response.

8.2.2.4 SIP Session Timer Expiry

On expiry of the SIP session timer on one of the connected CPM Client(s), if the CPM Participating Function is behaving as a B2BUA for the affected SIP session, the CPM Participating Function:

1. SHALL generate a SIP BYE request for that connected CPM Client(s) served by the CPM Participating Function for this SIP session according to the rules and procedures of [RFC3261];
2. If the action element <allow-offline-storage> is set to true, the CPM Participating Function:
   a. If a CPM Session History Folder is used, it SHALL create and store Group State Object in the CPM Session History Folder.
   b. SHALL include in the SIP BYE request the Message-UID header set to the retrieved UID information as defined in Appendix C “*CPM-defined Header fields*”.
3. SHALL send the SIP BYE request(s) according to the rules and procedures of [RFC3261] towards that connected CPM Client;
4. If the affected SIP session is being interworked, and if there are no other CPM Clients of the served CPM User connected into the SIP session, then the CPM Participating Function:
   a. SHALL generate SIP BYE requests for each of the associated IWFs according to the rules and procedures of [RFC3261];
   b. SHALL send these SIP BYE requests according to the rules and procedures of [RFC3261].
5. Otherwise, when the affected SIP session is not being interworked, the CPM Participating Function and if there are no other CPM Clients of the served CPM User connected into the SIP session, then:
   a. SHALL generate a SIP BYE request for the other CPM entity of the affected SIP session according to the rules and procedures of [RFC3261];
   b. SHALL send the SIP BYE request according to the rules and procedures of [RFC3261].
6. SHALL release all Media Plane resources associated with the SIP session according to the rules and procedures of [RFC3261] and [RFC4028], after a response was received for the SIP BYE requests.

If the CPM Participating Function behaves as a SIP proxy, it SHALL act as specified in [RFC4028].

8.2.2.5 Handle a CPM Session Modification Request

Upon receiving a SIP re-INVITE request from a served CPM Client, the CPM Participating Function:

1. SHALL act as specified in section 8.2.2.1 "*Handle a CPM Session Invitation*" except for step 4;
2. If the CPM Participating Function is a B2BUA:
    a. SHALL generate a SIP re-INVITE request according to the rules and procedures of [RFC3261]
    b. SHALL include the same URI identifying its own address that was used in the initial SIP INVITE request in the Contact header of the SIP re-INVITE request;
3. If the CPM Participating Function is acting as a SIP proxy, SHALL remove its own address from the Route header in the received SIP re-INVITE request according to the rules and procedures of [RFC3261];
4. If the CPM Session to be modified was established with IWF(s), it SHALL perform the following procedures:
    a. If the SIP re-INVITE request is removing existing Media Streams or changing characteristics of existing Media Streams, it SHALL send the SIP re-INVITE request to the IWF involved in that Media Stream;
    b. If the SIP re-INVITE request is adding a new Media Stream, it SHALL sequentially fork the SIP re-INVITE request to IWF(s) involved in the CPM Session until one of the IWFs responds with a SIP 200 "OK" response or all IWFs have been tried.
5. Otherwise, it SHALL send the SIP re-INVITE request according to the rules and procedures of the SIP/IP core.
6. If the service provider policy supports the direct delivery of originated CPM Chat Messages to all suitable CPM User's registered Clients and there are other CPM User's suitable CPM Clients connected to this CPM Session, the CPM Participating Function SHALL forward the SIP re-INVITE request to all other connected CPM Clients to this CPM Session, according to the rules and procedures of the SIP/IP core.

Upon receiving a SIP final response, the CPM Participating Function SHALL act as specified in section 8.2.2.1 "*Handle a CPM Session Invitation*".

Upon receiving a SIP 488 "Not Acceptable Here" response from all IWFs that the SIP re-INVITE request for adding a new Media Stream was sent to, the CPM Participating Function responsible for making interworking decision MAY, if service provider policy allows this, establish a new session towards an IWF for the new Media Stream; otherwise, the CPM Participating Function SHALL forward the SIP 488 "Not Acceptable Here" response according to the rules and procedures of the SIP/IP core.

In order to establish a new session towards an IWF for the new Media Stream, the CPM Participating Function SHALL generate a SIP INVITE request according to the rules and procedures of [RFC3261] with the following clarifications:

1. SHALL include a MIME SDP body for the new Media Stream as an SDP offer in the SIP INVITE request.
2. SHALL send the SIP INVITE to the ISF as described in section 6.5 "*Communicating with the ISF and IWF".*

Upon receiving a SIP ACK request, the CPM Participating Function SHALL forward the SIP ACK request to the SIP/IP core according to the rules and procedures of the SIP/IP core for delivery to the entity that accepted the session modification.

8.2.2.6 Handling of Participants Information

When a CPM Participating Function needs to obtain the Participants Information of a CPM Group Session, it subscribes to the CPM Controlling Function hosting the focus, using the conference event package described in [RFC4575].

The CPM Participating Function:

1. SHALL generate a SIP SUBSCRIBE request, according to the rules and procedures of [RFC6665] and [RFC4575];
2. SHALL set the Request-URI of the SIP SUBSCRIBE request to the CPM Group Session Identity;
3. SHALL include an Accept-Contact header with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' 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;
4. the CPM Participating Function SHALL set the P-Asserted-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group';
5. SHALL include a Contact header through which it can be reached according to [RFC3261] with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' 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;

6. SHALL populate the From header field with CPM Participating Function’s own SIP URI such as the configured CPMDeferredMsgMgmtURI;
7. SHALL set the P-Asserted-Identity header field to the SIP URI of the CPM User’s SIP URI;
8. SHALL send the SIP SUBSCRIBE request according to the rules and procedures of the SIP/IP core.

The response to the SIP SUBSCRIBE request SHALL be handled according to the rules and procedures of [RFC6665] and [RFC4575], and rules and procedures of the SIP/IP core.

When the CPM Participating Function needs to terminate the subscription, it SHALL proceed according to the rules and procedures of [RFC6665].

Note: The CPM Participating Function should keep locally the latest Participants Information received in the SIP NOTIFY to be delivered to CPM Client as part of the deferred delivery of the CPM Group Session to the CPM Client.

When functioning as a B2BUA for the SIP INVITE dialog (see e.g. section 8.2.2.1 “*Handle a CPM Session Invitation*”) the CPM Participating Function SHALL also receive the SIP SUBSCRIBE requests from the CPM Clients. The CPM Participating Function SHALL either forward this request to the CPM Controlling Function and terminate any subscription that it may have initiated itself or act as a B2BUA and initate its own subscription as described above or link it to an existing one. In both cases, the CPM Participating Function SHALL change the Request-URI value to the CPM Group Session Identity which it has received from the CPM Controlling Function. If multiple clients of the same CPM User can initiate subscriptions in parallel, the CPM Participating Function SHALL act as a B2BUA and ensure that towards the CPM Controlling Function only one subscription is active for that CPM User.

If there is no established session with the Controlling Function (i.e. when forwarding stored content), the CPM Participating Function MAY opt to not forward a SIP SUBSCRIBE request received from the client to the CPM Controlling Function. Instead it may return a SIP 200 OK response and generate an initial NOTIFY request to the client containing the Conference State information based on the locally stored Participants Information from the CPM Group Session.

Note: This is because according to [RFC3261] the CPM Participating Function has to provide an identity through which it can be reached in the Contact header field in that case and the CPM Client will use this address in the Request-URI of the SIP SUBSCRIBE request..

8.2.2.6.1 Receive Participant Information Notification

Upon receiving an incoming SIP NOTIFY request that is part of the same SIP dialog as a previously sent SIP SUBSCRIBE request for subscribing to participation information, the CPM Participating Function:

1. SHALL handle the request according to the rules and procedures of [RFC6665] and [RFC4575];
2. when receiving Subscription-State header to "terminated; reason=noresource" indicating that the CPM Group Session is now terminated or inactive, then:
   - if the action element includes an ‘<allow-store>’ sub-element set to “true”, or
   - a sub-element <allow-offline-storage> set to true,
3. then CPM Participating Function SHALL store the up-to-date Participant Information, <subject>, <icon> and <bulletin> values in a new Group State Object and/or a Conference Information Object in the CPM Message Store Server under the CPM Session folder. This will result in multiple Group State Objects, and/or a Conference Information Objects, each of which includes a snapshot state along the course of the CPM Session. The Group State Object with the latest timestamp includes the last state of the CPM Session.

   When storing Group State Object(s), a Group State Object SHALL be created by the CPM Participating Function from the coherent conference state. The Group State Object;

   a. SHALL include the "lastfocussessionid” attribute set to the CPM Group Session Identity from the <entity> attribute of the <conference-info> element of the conference event package,
   b. SHALL include the timestamp attribute set to current date and time,
   c. SHALL include the "group-type” attribute set to the type of CPM Group Session,
   d. SHALL include one "participant" element for each <user> element in the conference event package for which the <status> element is not set to “disconnected” with an <disconnection-method> element set to a value of “failed” or "departed", Each "participant" element:
      i. SHALL include the “comm-addr” attribute with the value taken from the <entity> attribute of the <user> element of the conference event package.
      ii. SHALL include the "name" attribute with the value taken from the <display-text> element of the <entity> attribute of the <user> element of the conference event package.

iii. SHALL include the "yourown" attribute if present in the <user> element of the conference event package.

iv. SHALL include the "role" attribute if the <role> element is present in the <user> element of the conference event package.

e. SHALL include the "subject" element if the <subject> element is present in the conference event package. The "subject" element

i. SHALL include in the "text" element the value of the <subject> element of the conference event package

ii. SHALL include the "participant" element with the value of <participant> element of the <subject-ext> element if present in the conference event package.

iii. SHALL the "timestamp" element with the value of <timestamp> element of the <subject-ext> element if present in the conference event package

f. SHALL include the "icon" element if the <icon> element is present in the conference event package. The "icon" element

i. SHALL include the source information from the <source> element of the conference event package which can contain either an icon-uri or file-info. If the <source> element of the conference event package is empty, then CPM Participating Function SHALL omit the source information in the Group State Object.

ii. SHALL include the "participant" element with the value of <participant> element of the <icon> element if present in the conference event package.

iii. SHALL the "timestamp" element with the value of <timestamp> element of the <icon> element if present in the conference event package.

g. SHALL include the "max-user-count" element if the <maximum-user-count> element is present in the <conference-description> of the conference event package.

h. SHALL include the "policy" element with the value of the <keywords> element, if present in the conference event package.

4. if for a CPM Session with a CPM Client Participant Information Notifications via Conference Information Object is applicable, SHALL generate a MSRP SEND request using the Content-Type type application/conference-info+xml and forward the body of the SIP NOTIFY;

5. SHALL forward the Participant Information in the dialogs that it has with the client(s) of the user, if any.

8.2.2.7 CPM Group Session Re-join Requests

8.2.2.7.1 General Re-join Handling

The CPM Participating Function SHALL determine that a re-join request was sent by a CPM Client when the received SIP INVITE has the following header fields as described in [RFC4579]:

a) Contains a Request-URI set to a CPM Group Session Identity,

b) The body does not include a Participants list (as opposed to a regular new CPM Group Session request which contains the participants list),

If the CPM Participating Function has service provider policies that determine whether the CPM Long-lived Group Session is supported or not, then the CPM Participating Function SHALL further handle it as described in sect. 8.2.2.7.2 "*CPM Long-lived Groups Session*". Otherwise, the CPM Participating Function SHALL forward the SIP request to the CPM Controlling Function once it received a user message or a "isComposing" message from the CPM Client.

If service provider policy supports the direct delivery of originated chat messages to all suitable CPM User's registered CPM Clients then the CPM Participating Function:

1) SHALL also initiate a re-join request to the other suitable registered CPM User's CPM Clients as described in section 8.2.2.1, if they are not already connected in the CPM Group Session.

The CPM Participating Function SHALL continue with procedures from section 8.2.2.7.2.

8.2.2.7.2 CPM Long-lived Group Session

A CPM Participating Function receiving a re-join SIP INVITE request for a CPM Group Session is considered to be on the originating side. However, when the CPM Group Session is still on-going and the CPM Participating Function is still in the path where it was added in a terminating role, it can be considered that even when CPM Client re-joins the CPM Group Session the CPM Participating Function remains in a terminating role, since the CPM Client is not the one that originated the CPM Group Session.

In either case, the procedures to be executed by the CPM Participating Function to re-join the CPM Client into the CPM Group Session are same regardless of whether it is considered to be on originating or terminating side.

Upon receiving a re-join SIP INVITE request from a CPM Client to a CPM Long-lived Group Session, the CPM Participating Function SHALL process the request as follows:

If the CPM Group Session is inactive:

1) If the CPM Client is allowed to re-start the CPM Long-lived Group Session which is now inactive, the CPM Participating Function SHALL act as specified in section 8.2.2.1 "*Handle a CPM Session Invitation*" and section 8.2.2.7.1 "*Re-join handling*";
2) If the CPM Client is not allowed to re-start the CPM Long-lived Group Session (e.g. subject to service provider policies), the CPM Participating Function SHALL reject the re-join INVITE with a SIP 403 "Forbidden" error code (e.g. if the CPM Controlling Function used by the CPM Client in the request is not allowed for re-starting CPM Long-lived Group Sessions, etc.) and SHALL include a SIP Warning header with the warning text set to "127 Service not authorised" in the response, as described in section 8.2.2.1 "*Handle a CPM Session Invitation*".

If the CPM Group Session is active, then:

1) If the CPM Participating Function remained in the media path (e.g. recording was enabled) and is already connected in the active CPM Group Session on behalf of the CPM User, then the CPM Participating Function SHALL re-connect the CPM Client into the on-going session as described in 8.2.2.7.1 "*Re-join handling*" and SHALL continue to have the role of a B2BUA. If the deferring of the missed CPM Group Session messages and disposition notifications is enabled for the CPM User, then the CPM Participating Function SHALL perform the deferred delivery of the CPM Group Session within this session, as described in section 0 "*Deferred CPM Group Session Handling*".
2) If the CPM Participating Function did not remain in the media path of the CPM Group Session, then it SHALL handle the request as described in sect. 8.2.2.1 "*Handle a CPM Session Invitation*".

8.2.2.8 Handling of Failed Deferred Delivery of Chat Messages

The CPM Participating Function in the originating network MAY retry delivery of failed 1-1 CPM Chat Messages to the recipient(s), subject to service provider policies.

8.2.3 CPM File Transfer Handling

8.2.3.1 Handle a CPM File Transfer

The CPM Participating Function SHALL handle a CPM File Transfer initiation request (i.e. a SIP INVITE request that includes the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer' in the Accept-Contact header field) in exactly the same manner as described for CPM Session Invitations in section 8.2.2.1 "*Handle a CPM Session Invitation*", with the following additional procedures before the CPM Participating Function decides to stay in the media path:

- The CPM Participating Function SHALL check if there is a rule in [OMA-XDM-Policy] in which the "enabler" attribute of the '<service>' sub-element inside the '<service-list>' element of the conditions part of the rule is set to "CPM" and the '<action>' element of the rule contains a '<max-file-transfer-size>' sub-element.
- If the <max-file-transfer-size> does not exist, or exists but has a value of zero, proceed with the remaining steps in section 8.2.2.1 "*Handle a CPM Session Invitation*";
- Otherwise, if the <max-file-transfer-size> exists, and is non-zero, and the "file-selector:size" attribute in the SDP exceeds that value, the CPM Participating function SHALL enforce the policy by returning a SIP 403 "Forbidden" response and SHALL include a SIP Warning header field with the warning text set to "133 Size exceeded" in the response according to the rules and procedures of [RFC3261].

The CPM Participating Function SHALL handle the request with the following differences from the procedures in section 8.2.2.1 "*Handle a CPM Session Invitation*":

a. If the CPM Participating Function is acting in a B2BUA role, then it SHALL include its own identity in the Contact header field and copy over any URI parameters from the Contact header field from the incoming SIP INVITE into the outgoing SIP INVITE request (e.g. "gr" parameter), else if the CPM Participating Function is acting in a proxy role it SHALL copy the Contact header field from the incoming SIP INVITE into the outgoing SIP INVITE request;
b. SHALL copy the Accept-Contact header field of the incoming SIP INVITE request to the outgoing SIP INVITE request, percent encoded as per [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI"*;
c. SHALL check if in the received SIP INVITE:
   i. the P-Preferred-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer'. In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add P-Asserted-Service header field and set it to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer'; or
   ii. the P-Preferred-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer.group'. In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add P-Asserted-Service header field and set it to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer.group'; or
   iii. the P-Asserted-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer' or the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer.group'. If another value is found, the CPM Participating Function MAY reject the SIP INVITE with a SIP 403 "Forbidden" response;
d. SHALL handle the SIP session timer parameters defined in [RFC4028], as specified in section 8.2.2.1 "*Handle a CPM Session Invitation*";
e. SHALL check the following SDP attributes received to determine if it is a new CPM File Transfer request, or if it is a CPM File Transfer pull to resume a previously interrupted CPM File Transfer, as follows:
   i. If a 'file-transfer-id' attribute was received and matches with a previously stored value, then the request is for a resumption operation and the CPM Participating Function SHALL continue with the procedures described in sect. 8.3.3.6 "*Receiving SIP INVITE request for deferred delivery of CPM File Transfer file(s)*";
   ii. If the SDP does not match, the CPM Participating Function SHALL respond with a SIP 488 "Not Acceptable Here" response with the correct SDP parameters.

When sending a SIP 200 "OK" response, the CPM Participating Functionr:

1) SHALL generate a SIP 200 "OK" response according to rules and procedures of [RFC3261];
2) SHALL include a Server header to indicate the OMA CPM release version of the CPM Participating Function as specified in Appendix D "Release Version in User-agent and Server headers";
3) SHALL include the SDP answer containing the file identities of the files as described in [RFC5547];
4) SHALL execute the Media Plane procedures as described in section 7.3.9.1 "*MSRP-based Media Streams*" to receive the files.

With respect to the SDP contents of outgoing SIP INVITE requests, the CPM Participating Function SHALL follow the rules and procedures of [RFC5547].

The MSRP media handling by the originating CPM Participating Function SHALL be done as described in section 8.6 "*Media Plane Handling*".

If interworking needs to be triggered for the CPM File Transfer delivery, the CPM Participating Function SHALL send the SIP INVITE request directly to the ISF, as described in section 6.5 "*Communicating with the ISF and IWF*".

8.2.3.2 Handle a CPM File Transfer Closing Request

The CPM Participating Function SHALL handle the closing of a CPM File Transfer session in exactly the same manner as defined for CPM Sessions in section 8.2.2.3 "*Handle a SIP BYE Request*" with the following clarifications.

1. if the action element <allow-offline-storage> is set to true for the originating CPM User:
   a. If the CPM File Transfer is not initiated in a CPM Session, the CPM Participating Function SHALL store the File Transfer history object as specified in section 8.5.3 "Recording CPM File Transfer"

   b. If the CPM File Transfer is initiated in a CPM Session, the CPM Participating Function SHALL store the File Transfer History Object in the CPM Session history folder associated with the CPM Session as specified in section 8.5.3 “Recording CPM File Transfer”, if a corresponding CPM Session history folder exists. If not, the CPM Participating Function SHALL create one with a new session info object in the CPM Session history folder.
   c. SHALL include in the SIP 200 “OK” response to the served CPM Client, a Message-UID header field set to the UID of the stored CPM File Transfer History object.
2. Otherwise, it SHALL return SIP 200 “OK” response without a Message-UID header field.

8.2.3.3 Handling of Failed Deferred Delivery of File Transfers

The CPM Participating Function in the originating network MAY retry delivery of failed 1-1 CPM File Transfer files to a recipient, subject to service provider policies.

If the destination address is:

– neither SIP URI address nor TEL URI address, the CPM Participating Function MAY send the SIP INVITE request directly to the ISF, as described in section 6.5 “*Communicating with the ISF and IWF*”;
– otherwise, the CPM Participating Function SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

8.2.4 Sending a Disposition Notification

The CPM Participating Function SHALL forward disposition notifications originated from one of the CPM entities to which the message was delivered (e.g. CPM IWF, CPM Client) via the SIP/IP core to the original sender of that CPM Message or CPM File Transfer request. In case the message was delivered to multiple endpoints, a similar caching strategy SHALL be followed as on the terminating side, described in section 8.3.4 “*Receiving a disposition notification*”.

If the disposition notification is a read report received from one of the served CPM User’s Clients, the CPM Participating Function SHOULD set the \Seen flag for that CPM Message object, or CPM File Transfer history object, in the CPM Message Store.

The CPM Participating Function SHALL NOT store the disposition notifications originated by the served CPM User Clients, or sent by network entities on behalf of the served CPM User.

8.2.4.1 Sending a Disposition Notification on behalf of a Served CPM User

The CPM Participating Function SHALL generate and send a disposition notification on behalf of the served CPM User in the following cases:

1. When failing to deliver a CPM Message(s), CPM File Transfer file(s) and a “negative-delivery” notification was requested by the originator of the CPM Message, or
2. When successfully delivering the CPM Message(s) or CPM File Transfer file(s) (including successful deferred delivery) to a CPM Client that does not support IMDN for that type of CPM Message or CPM File Transfer, e.g., as detected in the User-Agent header field (e.g. a CPM v1.0 client does not support IMDN for CPM Chat Messages), or via the SDP negotiation, or
3. When a CPM Chat Message is discarded after expiry according to user preferences and a “negative-delivery” notification was requested by the originator of the CPM Chat Message.

When having to generate and send a disposition notification on behalf of a recipient CPM Client, the CPM Participating Function SHALL generate a disposition notification as a receiving entity, as described in sect. 5.4.1 “*Generate Delivery Notification*”, with the following clarification:

- The CPM Participating Function SHALL set the appropriate <status> field within the <delivery-notification> element of the IMDN to the correct value (i.e. <delivered/> for a “positive-delivery” notification and <failed/> for a “negative-delivery” notification);
- When the CPM Participating Function determines that it needs to store the message, for which it sent an IMDN notification on behalf of the CPM User, then it SHALL follow the appropriate procedures from section 8.5 “*Record CPM Conversation History*” with the following clarifications:

  - SHALL remove the values "negative-delivery" and "positive-delivery" from the Disposition-Notification header field in CPIM header of the received CPM Standalone Message as defined in [RFC5438]. If these are the only values present, then it SHALL completely remove the Disposition-Notification header field from the CPIM headers stored.
- When an error occurs during the MSRP transmission of the original CPM File Transfer request, the CPM Participating Function SHALL send a SIP MESSAGE with the delivery notification <status> sub-element of “failed” directly to the original sender or of the CPM File Transfer.

For network optimization purposes, the aggregation of IMDNs as specified in [RFC5438] MAY be supported by the CPM Participating Function.

When IMDNs are sent to a remote network, the CPM Participating Function SHALL send the IMDN for each individual CPM Chat Message or CPM File Transfer file, leaving the target CPM Participating Function to handle them as per its own support and Service Provider policies (i.e. either aggregate them before delivery to clients, or deliver them individually).

8.3 Procedures in the Terminating Network

A CPM Participating Function in a terminating network handles incoming CPM communications in the following manners:

- Upon receiving a SIP MESSAGE request with the CPM Feature Tag set to one of:
    - ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg’, or
    - ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg’, or
    - ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’, or
    - ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer’,

  defined in Appendix H included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP MESSAGE request as described in section 8.3.1.1 “*Handle a Pager Mode CPM Standalone Message and SIP IMDNs*”. If any additional strings (e.g. IARIs) are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.
- Upon receiving a SIP MESSAGE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg’ defined in Appendix H, included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP MESSAGE request as described in section 6.7.3.2 “Receiving One Time CPM Events”.
- Upon receiving a SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg’ defined in Appendix H included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP INVITE request as described in section 8.3.1.2 “*Handle a Large Message Mode*”. If any additional strings (e.g. IARIs) are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.
- Upon receiving a SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred’ defined in Appendix H included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP INVITE request as described in section 8.3.1.6.3 “*Sending Notifications and Awaiting CPM Client Action*”. If any additional strings (e.g. IARIs) are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.
- Upon receiving a SIP SUBSCRIBE request with the Event header set to ’deferred-messages’, the CPM Participating Function SHALL handle this SIP SUBSCRIBE request as described in section 8.3.1.6.4 “*Delivering Deferred CPM Messages to the Message Storage Server*”. If any additional strings (e.g. IARIs) are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.

- Upon receiving a SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' defined in Appendix H included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP INVITE request as described in section 8.3.2.1 "*Handle a CPM Session Invitation*". If any additional strings (e.g. IARIs) are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.
- Upon receiving a SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer' defined in Appendix H included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP INVITE request as described in section 8.3.3.1 "*Handle a CPM File Transfer Initiation*". If any additional strings (e.g. IARIs) are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.
- Upon receiving a SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg' defined in Appendix H included in the Accept-Contact header field, the CPM Participating Function SHALL handle this SIP INVITE request as:
  - described in section 6.7.3.2 "*Receiving One Time CPM Events*", when the SDP directional media attribute is set to a=sendonly, or
  - described in section 6.7.3.4 "*Receiving a Bi-directional Session for CPM Events Invitation*", when the SDP directional media attribute is set to a=sendrecv.

8.3.1 CPM Standalone Message Handling

8.3.1.1 Handle a Pager Mode CPM Standalone Message and SIP IMDNs

Upon receiving a SIP MESSAGE request with one of the CPM Feature Tag values listed in section 8.3 "*Procedures in the Terminating Network*" included in the Accept-Contact header, the CPM Participating Function:

1. If the CPM Participating Function requires a specific User Agent version to be supported, the CPM Participating Function SHALL check the "User Agent" header field to determine if the CPM Participating Function supports the User Agent version and if not, the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "132 Version not supported" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with rest of the steps;
2. If the CPM Participating Function does not allow anonymity and anonymity is requested, the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "119 Anonymity not allowed" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
3. MAY heck if the Authenticated Originator CPM Address of the SIP MESSAGE request is included in the standalone messages Blacklist URI-list stored in [OMA-XDM-List]. If it is, then the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "122 Function not allowed" according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
4. MAY check in the recipient's user preferences retrieved from XDMS as described in 8.4.1 "*Retrieving User Preferences*" and if there is a rule in [OMA-XDM-Policy] in which the "enabler" attribute of the '<service>' sub-element inside the '<service-list>' element of the conditions part of the rule is set to "CPM" and the '<media-list>' element of the conditions part of the rule contains the '<standalone-message>' element. If such a rule exists then the CPM Participating Function SHALL handle the value of the '<action>' element as follows:
   a. If the action element includes an '<allow-reject-invite>' sub-element set to "true", the CPM Participating Function SHALL return a SIP 403 "Forbidden" response to the CPM Client. Also the CPM Participating Function SHALL include a SIP Warning header with the warning text set to "122 Function not allowed" according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

  b. If the action element includes an '<allow-do-not-disturb>' sub-element set to "true". the CPM Participating Function SHALL defer the message as specified in section 8.3.1.6 "*Defer CPM Standalone Messages*" and SHALL return a SIP 202 "Accepted" response.

     Otherwise, continue with the rest of the steps.

  c. If the action element includes an '<allow-interwork>' sub-element set to "true" or if the action element includes an '<allow-deliver-and-interwork>' sub-element set to "true", the CPM Participating Function SHALL send the SIP MESSAGE directly to the ISF as described in section 6.5 "*Communicating with the ISF and IWF*".

  d. If the action element includes an '<allow-store>' sub-element set to "true", the CPM Participating Function:

     i. SHALL store the message in the user's message store in the Message Storage Server according to procedures specified in section 6.3.1 "*Object Store Operation*" of [OMA-CPM_TS_MessageStorage].

     ii. SHALL return a SIP 200 "OK" response.

     Otherwise, continue with the rest of the steps;

  e. If the action element includes an '<allow-forward>' sub-element set to "true", the CPM Participating Function SHALL change the address of the recipient to the one provided in the user preferences only if the Content-Type of the SIP MESSAGE is not set to "message/imdn+xml", and send the CPM Standalone Message to that address through the SIP/IP core. If the Content-Type of the SIP MESSAGE is set to "message/imdn+xml", then the CPM Participating Function SHALL NOT change the address of the recipient.

     Otherwise, continue with the rest of the steps;

  f. If the action element includes an '<allow-defer>' sub-element set to "true", the CPM Participating Function SHALL defer the message as specified in section 8.3.1.6 "*Defer CPM Standalone Messages*" and SHALL return a SIP 202 "Accepted" response;

     Otherwise, continue with the rest of the steps;

5. If in step 4 no rule matched or if the action element of the matching rule included an '<allow-deliver-and-interwork>' sub-element set to "true" or an '<allow-interwork>' sub-element set to "true", the CPM Participating Function MAY check if there is a rule in [OMA-XDM-Policy] in which the "enabler" attribute of the '<service>' sub-element inside the '<service-list>' element of the conditions part of the rule is set to "CPM" and the '<action>' element of the rule contains an '<allow-offline-storage>' sub-element set to "true". If such a rule exists, it SHALL execute the processing described in section 8.5.1 "*Record CPM Conversation History".*

6. If the Content-Type of the SIP MESSAGE is "message/imdn+xml", the CPM Participating Function SHALL send the request via the SIP/IP core towards the user's CPM Client, and not execute the remaining steps.

7. If in step 4 no rule matched, or if the action element of the matching rule included the '<allow-deliver-and-interwork>' sub-element, the CPM Participating Function SHALL continue with the following steps:

8. SHALL determine which of the registered CPM Clients are expected to receive the CPM Standalone Message as follows:

  a. MAY check for rules in [OMA-XDM-Policy] satisfying the following criteria:

     1) The <service-list> element inside the <conditions> element contains a <service> element with the attribute "enabler" set to "CPM".

     2) The <upp-list> element inside the <conditions> element contains the active User Preference Profile name of one of the registered CPM Clients of the recipient CPM User.

     3) Other elements inside the <conditions> element evaluate to true for the CPM Standalone Message.

     4) The <action> element contains a <allow-reject-invite> element, as defined in [OMA-XDM-Policy], and with its value set to "true".

     5) SHALL exclude each registered CPM Client whose active User Preference Profile is mentioned in one of the rules found as a result of step i from the list of registered CPM Clients that will receive the CPM Standalone Message;

b. SHALL evaluate other information sources to determine device connectivity over CS or IP access and, if evaluated, SHALL exclude those registered CPM Clients whose connectivity does not allow to receive the CPM Standalone Message , as follows:

   i. SHALL further exclude those registered CPM Clients whose capabilities (learnt via the registration NOTIFY request as specified in section 8.1.4 "*Using the Registration Event Information*", or via other information sources) do not allow to receive the CPM Standalone Message.

   ii. SHALL further exclude those registered CPM Clients that should not receive the CPM Standalone Message according to service provider policies;

   iii. MAY evaluate whether or not any one of the registered CPM Client resides on a Primary Device (as defined in section 6.6) based on the data learnt via the registration NOTIFY request as specified in section 8.1.2 "*Receive Registration Event Information Notifications*".

NOTE: steps ii, iii, iv, and v implement the notion of a CPM Client being "suitable" (or not) for a request as described in section 6.6 "Suitable CPM Clients". This notion of "suitable" applies equally to other requests, i.e. Large Message Mode CPM Standalone Message request, CPM Session Invitation request and CPM File Transfer request, see affected sections.

9. SHALL generate a SIP MESSAGE request according to the rules and procedures of [RFC3428]:

   i. for each registered CPM Client selected in step 8 above;

   ii. If service provider policies require that the message is delivered immediately to the Primary Device and none of the registered suitable CPM Clients is on the Primary Device, then the CPM Participating Function SHALL also deliver the CPM Message using a Non-CPM Communication Service to the CPM Client on the Primary Device. To this purpose, the CPM Participating Function SHALL send the SIP MESSAGE directly to the ISF as described in section 6.5 "*Communicating with the ISF and IWF*";

   iii. to be sent directly to the ISF as described in section 6.5 "*Communicating with the ISF and IWF*" based on service provider policy if there is no suitable registered CPM Client which resides on a Primary Device as determined in step 88.b)iii. and there is no rule matched in step 4b;

10. SHALL copy the Accept-Contact header(s) of the incoming SIP MESSAGE request to the outgoing SIP MESSAGE request (discarding the Accept-Contact header with the sip.instance tag and value for the device identifier, if one was provided);

11. MAY replace each Media Object attached to the CPM Standalone Message with a reference, as described in section 8.3.1.4 "*Replacing Media with a Reference*";

12. SHALL check if in the received SIP MESSAGE for Pager Mode CPM Standalone Message:

    a) the P-Preferred-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg'. In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add P-Asserted-Service header field and set it to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg'; or

    b) the P-Preferred-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.group'. In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add P-Asserted-Service header field and set it to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.group'; or

    c) if P-Asserted-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg' or the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.group'. If another value is found, the CPM Participating Function MAY reject the SIP INVITE with a SIP 403 "Forbidden" response;

13. SHALL include a Request-URI and set its value to the suitable CPM Client's identity, which is one of the public GRUU if one was provided, or the recipient CPM User's authenticated address obtained in the registration NOTIFY request as specified in the subsection 8.1.2 "*Receive Registration Event Information Notifications*";

14. If the CPM Standalone Message was recorded in step 5, the CPM Participating Function SHALL include the UID information received in section 8.5.1 "*Record CPM Conversation History*" in the Message-UID header as specified in Appendix C.1.6 "*Message-UID*";

15. SHALL send the request via the SIP/IP core towards each user's selected CPM Client.

If no suitable registered CPM Clients are found to receive the Pager Mode CPM Standalone Message request, then the CPM Participating Function SHALL follow procedures in the section 8.3.1.3 "*Applying delivery policies*" and stop with this procedure.

With the exception of message forwarded case upon receiving the first SIP 2xx final response from one of the selected CPM Clients, the CPM Participating Function SHALL send the SIP final response along the signalling path towards the originating CPM Client according to the rules and procedures of [RFC3261].

The CPM Participating Function SHOULD determine if it needs to send a delivery notification on behalf of the CPM User, when one was requested by the originator, as described in section 8.2.4.1 "*Sending a Disposition Notification on behalf of a Served CPM User*".

8.3.1.2 Handle a Large Message Mode CPM Standalone Message

Upon receiving a SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg' included in the Accept-Contact header corresponding to Large Message Mode CPM Standalone Message, the CPM Participating Function SHALL execute the following:

1. If the CPM Participating Function requires a specific User Agent version to be supported, the CPM Participating Function SHALL check the "User Agent" header field to determine if the CPM Participating Function supports the User Agent version and if not, the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "132 Version not supported" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with rest of the steps;
2. If the CPM Participating Function does not allow anonymity and anonymity is requested, the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "119 Anonymity not allowed" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
3. If the "Session-Expires" header is included, the CPM Participating Function SHALL handle the header field according to procedures of [RFC4028].
4. SHALL behave as a B2BUA according to the rules and procedures of [RFC3261] for the duration of the session;
5. MAY check if the Authenticated Originator CPM Address of the SIP INVITE request is included in the standalone message Blacklist URI-list stored in [OMA-XDM-List]. If it is, then the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "122 Function not allowed" according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
6. MAY check in the recipient's user preferences retrieved from XDMS as described in 8.4.1 "*Retrieving User Preferences*" and if there is a rule in [OMA-XDM-Policy] in which the "enabler" attribute of the '<service>' sub-element inside the '<service-list>' element of the conditions part of the rule is set to "CPM" and the '<media-list>' element of the conditions part of the rule contains the '<standalone-message>' element. If such a rule exists then the CPM Participating Function SHALL handle the value of the '<action>' element as follows:
   a. If the action element includes an '<allow-reject-invite>' sub-element set to "true", the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "122 Function not allowed" in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;
   b. If the action element includes an '<allow-do-not-disturb>' sub-element set to "true". the CPM Participating Function SHALL defer the message as specified in section 8.3.1.6 "*Defer CPM Standalone Messages*" and SHALL return a SIP 202 "Accepted" response.

      Otherwise, continue with the rest of the steps.
   c. If the action element includes an '<allow-interwork>' sub-element set to "true" or if the action element includes an '<allow-deliver-and-interwork>' sub-element set to "true", the CPM Participating Function:
      i. SHALL generate a SIP INVITE request for the ISF according to the rules and procedures of [RFC3261];

ii. SHALL copy the Conversation-ID, Contribution-ID, InReplyTo-Contribution-ID (if present), Contact and Accept-Contact header fields of the incoming SIP INVITE request to the outgoing SIP INVITE request, as well as any Subject and recipient-list-history body from the original SIP INVITE request;

iii. SHALL include a Supported header field with the option tag 'timer' and a 'recipient-list-invite' tag if included in the original SIP INVITE request;

iv. SHALL set the refresher parameter to "uac" in the Session-Expires header field;

v. SHALL set the delta-seconds value to a value lower than 1800 in the Session-Expires header according to service provider policy and to the rules and procedures of [RFC4028];

vi. SHOULD NOT include the Min-SE header field according to the rules and procedures of [RFC4028];

vii. SHALL check if in the received SIP INVITE:

   1. the P-Preferred-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg'. In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add P-Asserted-Service header field and set it to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.largemsg', or

   2. the P-Preferred-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group', In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add P-Asserted-Service header field and set it to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.largemsg.group', or

   3. if P-Asserted-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg'. or

   4. if P-Asserted-Service header field is present and carries the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group'.

   If another value is found in either of the two header fields, the CPM Participating Function MAY reject the SIP INVITE with a SIP 403 "Forbidden" response;

viii. SHALL copy the Contact header field of the received SIP INVITE, if the 'isfocus' feature parameter as described in Appendix C.1.7 "Contact header field values" is present, otherwise SHALL include a Contact header field with a URI identifying its own address;

ix. SHALL include a User-Agent header to indicate the OMA CPM release version of the Participating Function as specified in Appendix D "*Release Version in User-agent and Server headers*";

x. SHALL include the MIME SDP body received in the SIP INVITE request as a SDP offer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714] with the following clarification, the CPM Participating Function:

   1. SHALL include media line proposing MSRP media parameters;
   2. SHALL populate its own MSRP URI as a=path:MSRP URI.
   3. SHALL set the SDP directional media attribute to a=sendonly;
   4. SHALL set the size as a=file-selector:size:actual message size;
   5. SHALL set the a=setup attribute as "actpass";

xi. SHALL send the SIP INVITE request directly to the ISF as described in section 6.5 "*Communicating with the ISF and IWF".*

d. If the action element includes an '<allow-store>' sub-element set to "true", the CPM Participating Function:

   i. SHALL establish an MSRP session to receive the message as described in 8.3.1.5 "*Establish MSRP Session for Receiving Large Message Mode*";

   ii. SHALL store the message in the Message Storage Server according to procedures specified in section 6.3.1 "*Object Store Operation*" of [OMA-CPM_TS_MessageStorage] and obtain an UID from the Message Storage Server.

   Otherwise, continue with the rest of the steps;

e. If the action element includes an '<allow-forward>' sub-element set to "true", the CPM Participating Function SHALL change the address of the recipient to the one provided in the user preferences and send the INVITE to that address through the SIP/IP core.

   Otherwise, continue with the rest of the steps;

f. If the action element includes an '<allow-defer>' sub-element set to "true", the CPM Participating Function:

   i. SHALL establish an MSRP session to receive the message as described in section 8.3.1.5 "*Establish MSRP Session for Receiving Large Message Mode*";

   ii. SHALL defer the message as described in section 8.3.1.6 "*Defer CPM Standalone Messages*".

   Otherwise, continue with the rest of the steps;

7. If in step 6 no rule matched or if the action element of the matching rule included an '<allow-deliver-and-interwork>' sub-element set to "true", the CPM Participating Function SHALL continue with the following steps:

   a. SHALL determine which of the CPM Clients are expected to receive the CPM Standalone Message by evaluating which CPM Clients are suitable, as described in step 8 of section 8.3.1.1 "*Handle a Pager Mode CPM Standalone Message*";

   b. SHALL generate a SIP INVITE request for each selected suitable registered CPM Client according to the rules and procedures of [RFC3261]. If no suitable registered CPM Clients are found to receive the request, then the CPM Participating Function SHALL follow procedures in the section 8.3.1.3 "*Applying delivery policies*" and stop with this procedure;

   c. MAY generate a SIP INVITE request to be sent directly to the ISF as described in section 6.5 "*Communicating with the ISF and IWF*" based on service provider policy if there is no CPM Client registered which resides on a Primary Device as determined in step 8 of section 8.3.1.1 "*Handle a Pager Mode CPM Standalone Message*" and there was no rule matched in step 6c. The SIP INVITE request is generated according to the rules and procedures of [RFC3261];

   d. SHALL copy any recipient-list-history body that was included in the received SIP INVITE request;

   e. SHALL copy any Subject, Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID SIP header fields from the received INVITE request;

   f. SHALL include a Supported SIP header field with the option tag 'timer' and if included in the received SIP INVITE request a 'recipient-list-invite' tag;

   g. SHALL set the refresher parameter to "uas" in the Session-Expires header field;

   h. SHALL set the delta-seconds value to a value lower than 1800 in the Session-Expires header field according to service provider policy and to the rules and procedures of [RFC4028];

   i. SHOULD NOT include the Min-SE header field according to the rules and procedures of [RFC4028];

   j. SHALL copy the Contact header and Accept-Contact header of the incoming SIP INVITE request to the outgoing SIP INVITE request;

   k. SHALL copy the Contact header field of the received SIP INVITE, if the 'isfocus' feature parameter as described in Appendix C.1.7 "Contact header field values" is present, otherwise SHALL include a Contact header with a URI identifying its own address;

   l. SHALL include a Request-URI header and set its value to the suitable CPM Client's identity, which is one of, public GRUU or the recipient CPM User's authenticated address obtained in the registration NOTIFY request as specified in the subsection 8.1.2 "*Receive Registration Event Information Notifications*";

   m. SHALL include the CPM release version in the User-Agent header as specified in Appendix D "*Release Version in User-agent and Server headers*";

   n. SHALL include the MIME SDP body received in the SIP INVITE request as a SDP offer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714] with the following clarification, the CPM Participating Function:

      i. SHALL include media line proposing MSRP media parameters;

      ii. SHALL supply its own MSRP URI as a=path:MSRP URI;

      iii. SHALL set the SDP directional media attribute to a=sendonly;

      iv. SHALL set the size as a=file-selector:size:actual message size;

      v. SHALL set the a=setup attribute as "actpass" according to [RFC6135];

vi. If not configured for MSRP interoperability, or if no MSRP interoperability is needed, it SHALL include an SDP 'msrp-cema' attribute in the MSRP media description of the SDP. Otherwise, it SHALL proceed according to section 5.2.1.4 “*Handling of Media connection parameters for MSRP*”.

o. SHALL send the request via the SIP/IP core towards each selected registered CPM Client and, if applicable, directly to the ISF as described in section 6.5 “*Communicating with the ISF and IWF*”.

Upon receiving the first SIP 200 “OK” response returned from the SIP INVITE requests delivered to the CPM Clients and/or ISF, the CPM Participating Function SHALL establish an MSRP session to receive the message as described in 8.3.1.5 “*Establish MSRP Session for Receiving Large Message Mode CPM Standalone Message”*.

If the recipient’s user preferences retrieved from XDMS as described in section 8.4.1 “*Retrieving User Preferences*” contain a rule in [OMA-XDM-Policy] in which the “enabler” attribute of the ‘<service>’ sub-element inside the ‘<service-list>’ element of the conditions part of the rule is set to “CPM” and the ‘<action>’ element of the rule contains an ‘<allow-offline-storage>’ sub-element set to “true”, the CPM Participating Function SHALL execute the processing described in section 8.5.1 “*Record CPM Conversation History*”.

Upon receiving a SIP 200 “OK” response from a CPM Client or from the ISF, the CPM Participating Function:

1. SHALL start listening for the incoming MSRP session and send a SIP ACK request to the sender of the SIP 200 “OK” response according to the rules and procedures of the [RFC3261].
2. SHALL follow the actions according to the rules and procedures of [RFC3261], [RFC4975], [RFC6135] and [RFC6714] to set up an MSRP session with the sender of the SIP 200 “OK” response;
3. SHALL send one or more MSRP SEND request(s) according to rules and procedures of [RFC4975] taking into account the maximum chunk size negotiated according to section 5.2.1, if any, to transfer the Large Message Mode CPM Standalone Message;
4. SHALL generate a SIP BYE request and, if user preference was set to store the CPM Conversation History, the CPM Participating Function SHALL include the Message-UID header set to the retrieved UID information as defined in Appendix C “*CPM-defined Header fields*”.
5. SHALL send a SIP BYE request towards the sender of the SIP 200 “OK” response according to the rules and procedures of the [RFC3261];
6. SHALL release the resources related to the recipient after the delivery.

   NOTE: These steps will be performed for each SIP 200 “OK” received by the CPM Participating Function.

Upon receiving a SIP BYE request from the originating side, the CPM Participating Function

1. SHALL send a SIP 200 “OK” response to the originating side;
2. SHALL release the resources related to the session on the originating side.

Upon receiving a SIP BYE request from a CPM client or the ISF, the CPM Participating Function

1. SHALL send a SIP 200 “OK” response to the sender of the SIP BYE request;
2. SHALL release the media resources related to the session with the sender of the SIP BYE request;
3. SHALL verify whether a session is established to another CPM client of the user or the ISF or whether such a session is pending;
4. SHALL, if no session is established or pending, generate and send a SIP BYE request to the originating side;
5. SHALL, if a SIP BYE request was sent tot the originating side, release the media resources related to the session on the originating side on receiving a SIP final response to the SIP BYE request ;

The CPM Participating Function SHOULD determine if it needs to send a delivery notification on behalf of the CPM User, when one was requested by the originator, as described in section 8.2.4.1 “*Sending a Disposition Notification on behalf of a Served CPM User*”.

8.3.1.3 Applying delivery policies

Terminating delivery behaviour is subject to service provider policies. If :

– no suitable CPM Clients are found (see section 6.6 “*Suitable CPM Clients*” for a description of what it means for a CPM client to be suitable and registered), or

- none of the suitable registered CPM Clients, nor, if applicable the ISF, return a SIP 200 “OK” response, or
- if an MSRP SEND error response is received from all selected CPM Clients and/or, if applicable, from the ISF (i.e. if the message was not delivered),

the CPM Participating Function SHALL check the service provider policy about handling undelivered messages, as described in section 8.3.6 “*Delivery Policies in Terminating CPM Participating Function*”.

8.3.1.4 Replacing Media with a Reference

The CPM Participating Function SHALL replace all Media Objects attached to a CPM Standalone Message with a reference if there exists a rule in [OMA-XDM-Policy] satisfying the following criteria:

1. The <service-list> element inside the <conditions> element contains a <service> element with the attribute “enabler” set to “CPM”;
2. The <action> element contains a <allow-deliver-reference-media> element, as defined in [OMA-XDM-Policy], and with its value set to “true”.

If such a rule is found, the CPM Participating Function:

1. SHALL store the CPM Standalone Message to the Message Storage Server according to procedures specified in section 6.3.1 “*Object Store Operation*” of [OMA-CPM_TS_MessageStorage] and obtain an UID from the Message Storage Server;
2. SHALL fetch the body structure for the stored CPM Standalone Message corresponding to the UID from the Message Storage Server according to procedures of [RFC3501];
3. SHALL generate IMAP URL for each Media Object decided to be replaced according to procedures of [RFC5092];

Once references to each Media Object are obtained, the CPM Participating Function

1. SHALL remove the Media Objects to be replaced from the received CPM Standalone Message as indicated in the active User Preference Profile;
2. SHALL include the IMAP URL in the body of the CPM Standalone Message corresponding to each Media Object to be replaced according to procedures of [RFC4483].

   NOTE: If there are other Content-Type headers within the CPM Standalone Message, the CPM Participating Function keeps the Content-Type header “multipart/mixed” with the proper boundary value to distinguish the URL part and existing parts as described in [RFC2046].

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

To establish an MSRP session with the incoming leg, the CPM Participating Function:

1. If a final SIP response was not yet sent, it SHALL generate a SIP 200 “OK” response according to the rules and procedures of [RFC3261] , to a received SIP INVITE described in section 8.3.1.2 “*Handle a Large Message Mode CPM Standalone Message*“; as follows:
   a. SHALL copy the Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID (if present) header fields from the original SIP INVITE request;
   b. SHALL include a Supported header field with the option tag ‘timer’; and
   c. If there was a Session-Expires header field in the original SIP INVITE request, SHALL copy the Session-Expires header field from the original SIP INVITE request according to the rules and procedures of [RFC4028];
   d. If there was no Session-Expires header field in the original SIP INVITE request, SHALL include a Session-Expires header field with the refresher parameter set to “uas” and set the delta-seconds value to a value lower than 1800 in the Session-Expires header according to service provider policy and to the rules and procedures of [RFC4028];
   e. SHALL include the SDP received in the response of SIP INVITE request as answer SDP according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714] with the following clarification, CPM Participating Function:
      i. SHALL include media line proposing MSRP media parameters;
      ii. SHALL populate its own MSRP URI as a=path:MSRP URI;

iii. SHALL set the SDP directional media attribute to a=recvonly;

iv. SHALL set the a=setup attribute as "passive";

v. If the CPM Participating Function is configured to apply MSRP interoperability according to section 5.2.1.4 "*Handling of Media connection parameters for MSRP session matching*", and the SDP received on this CPM leg did not include an SDP 'msrp-cema' attribute, the CPM Participating Function SHALL follow the procedures specified in section 5.2.1.4.1 "*Legacy MSRP session matching*" for the corresponding SDP parameters in the SIP response;

f. SHALL include a Server header field to indicate the OMA CPM release version of the CPM Participating Function as specified in Appendix D "*Release Version in User-agent and Server headers*";

g. SHALL include a Contact header field with a URI identifying its own address;

2. SHALL act as a "passive" endpoint, according to [RFC6135] to establish an MSRP connection;
3. SHALL send the SIP 200 "OK" response towards the incoming leg according to procedures of SIP/IP core.

Upon receiving a SIP CANCEL request, the CPM Participating Function SHALL act as a UAS to handle the SIP CANCEL request according to the rules and procedures of [RFC3261] and SHALL forward the SIP CANCEL request to the other leg i.e. to all CPM entities involved in the on-going SIP session (e.g. CPM Client(s) and/or the ISF, as applicable).

Upon receiving a SIP ACK request, the CPM Participating Function SHALL act as a "passive" endpoint, according to [RFC6135] and SHALL establish an MSRP connection as follows:

A. If the SDP offer received on the incoming leg:

1) contained the SDP 'msrp-cema' attribute, the CPM Participating Function SHALL apply the connection procedures according to [RFC6714];

2) Otherwise, if the SDP offer did not contain the SDP 'msrp-cema' attribute:

i. If the CPM Participating Function is not configured to apply MSRP interoperability according to section 5.2.1.4 "*Handling of Media connection parameters for MSRP*", the CPM Participating Function SHALL apply the connection procedures according to [RFC4975].

ii. Otherwise, if the CPM Participating Function is configured to apply MSRP interoperability according to section 5.2.1.4 "*Handling of Media connection parameters for MSRP*", the CPM Participating Function SHALL apply MSRP session matching as described in section 5.2.1.4.1 "*Legacy MSRP session matching*".

Upon receiving an MSRP SEND request, the CPM Participating Function:

1. SHALL wait if applicable, until all chunks of the message have been received via subsequent MSRP SEND requests and SHALL reassemble the message from these chunks for further processing, i.e. either for:

   a. re-chunking purposes according to the rules and procedures of [RFC4975]. If chunking negotiation is used, it SHALL take into account the maximum MSRP chunk size negotiated as specified in section 5.2.1 1. "*SDP Contents when Initiating or Modifying a CPM Session*"; or for

   b. storage purposes, in case the delivery will be successful; or for

   c. alternative delivery methods as per service provider policies, in the case of failure to deliver the CPM Large Message;

2. SHALL generate and send one or more MSRP 200 "OK" responses back, one per MSRP SEND request received, according to the rules and procedures of [RFC4975];
3. SHALL send the MSRP SEND request(s) to the served CPM Client(s) carrying the message body received, following the procedures described in section 8.6.2 "*Procedures to Send MSRP";*

   NOTE: depending on operator policy (e.g. to fulfil screening requirements on the content), the CPM Participating Function may either start generating and sending the MSRP SEND requests as soon as all CPIM headers are received from the originating side or after re-assembly of the received message is complete.

4. Upon receiving the MSRP responses from the served CPM Client(s), it SHALL use the assembled CPM Large Message for either:

   a. storage purposes, once the CPM Large Message was successfully delivered to at least one served CPM Client(s) and if the CPM Large Message must be stored in the CPM Message Store (e.g. recording of CPM Large Message is enabled); or

   b. to apply the delivery policies set by the service provider, in case of failure to successfully deliver the complete CPM Large Message to at least one of the served CPM Client(s). In this case, the CPM participating Function SHALL follow the procedures described in section 8.3.6 "*Delivery Policies in Terminating CPM Participating Function*".

;Upon receiving a SIP BYE request from the originating CPM User, CPM Participating Function SHALL respond to the SIP BYE request as described in [RFC3261].

8.3.1.6 Defer CPM Standalone Messages

Each time the CPM Participating Function receives a message that is to be deferred on behalf of the served CPM User, the CPM Participating Function SHALL check if there is a rule in [OMA-XDM-Policy] in which the "enabler" attribute of the '<service>' sub-element inside the '<service-list>' element of the conditions part of the rule is set to "CPM" and the '<media-list>' element of the conditions part of the rule contains an '<deferred-messages>' element. If such a rule exists then the CPM Participating Function SHALL check the value of the '<action>' element of that rule as follows:

1. If the '<action>' element includes an '<allow-push>' sub-element set to "true", the CPM Participating Function SHALL store the Deferred CPM Message in the Deferred CPM Message queue as described in section 8.3.1.6.1 "*Storing a Deferred CPM Message in the Deferred CPM Message Queue*" and SHALL handle the Deferred CPM Messages as described in section 8.3.1.6.2 "*Pushing Deferred CPM Messages*".
2. If the '<action>' element includes an '<allow-pull>' sub-element set to "true", the CPM Participating Function store the Deferred CPM Message in the Deferred CPM Message queue as described in section 8.3.1.6.1 "*Storing a Deferred CPM Message in the Deferred CPM Message Queue*" and SHALL handle the Deferred CPM Messages as described in section 8.3.1.6.3 "*Sending Notifications and Awaiting CPM Client Action*".
3. If the '<action>'element includes an '<allow-store>' sub-element set to "true", the CPM Participating Function SHALL handle the Deferred CPM Messages as described in section 8.3.1.6.4 "*Delivering Deferred CPM Messages to the Message Storage Server*".

   NOTE: The behaviour of the CPM Participating Function when none of the three options above have been enabled is implementation specific.

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

For each CPM Standalone Message that needs to be stored in the Deferred CPM Message queue, the CPM Participating Function:

1. SHALL, in case of a Pager Mode CPM Standalone Message, store the following information:
   a. The SIP MESSAGE request start-line, as well the following headers of SIP MESSAGE request:
      - From
      - To
      - P-Asserted-Identity, if there is and only if the sending User did not request anonymity
      - Subject (if present)
      - Date (added with current date and time if not present)
      - Accept-Contact
      - P-Asserted-Service or P-Preferred-Service, whichever is received
      - Content-Type (set to Content-Type of the SIP MESSAGE request message body, or to multipart/mixed if there is a recipient-uri-list as well as SIP MESSAGE request message body)
      - Content-Length:<messagesize>
      - cpm.Payload-Type, if present
      - Conversation-ID
      - Contribution-ID
      - InReplyTo-Contribution-ID (if present)

      NOTE: Some of the header fields shall not be stored, in particular header fields related to charging that are expected to be removed by the network.

   b. All CPIM headers relevant for IMDN
   c. body part consisting of all the recipients carried in recipient-uri-list body of the SIP MESSAGE request if present

   d. The complete message body or bodies of the SIP MESSAGE request;
2. SHALL, in the case of Large Message Mode CPM Standalone Message, store the following information:
   a. The SIP INVITE request start-line, as well as the following headers of SIP INVITE request of the Deferred CPM Message:
      - From
      - To
      - P-Asserted-Identity, if there is and only if the sending User did not request anonymity
      - Subject (if present)
      - Date (added with current date and time if not present)
      - Accept-Contact
      - Contact
      - P-Asserted-Service or P-Preferred-Service, whichever is rceived
      - Referred-By if present
      - Content-Type (set to Content-Type of the MSRP SEND, or to multipart/mixed if there is a recipient-uri-list as well as an MSRP SEND message)
      - Content-Length
      - Conversation-ID
      - Contribution-ID
      - InReplyTo-Contribution-ID (if present)
   b. body part consisting of all the recipients carried in recipient-uri-list body of the SIP INVITE request if present;
   c. the SDP body;
   d. The MSRP request start-line, as well as the relevant headers of the MSRP SEND request:
      - Message-ID,
      - Success-Report and Failure-Report if present, and
      - Content-Type
      - Byte-Range: 1-*/<messagesize>.
      - Content-Disposition
      - cpm.Payload-Type, if present
      - The complete message body. If the message arrives in more than one MSRP SEND request since it had been sent in chunks as defined in [RFC4975], then the CPM Participating Function shall assemble the message and store it as one message.
3. SHALL assign a unique message-URI-ID for each Deferred CPM Message whose general format is messageURI-ID =“sip:String@hostname” with the following clarification:
   a. The CPM Participating Function generates the unique string by itself
4. SHALL store the Deferred CPM Message in the Deferred CPM Message queue;
5. SHALL associate an expiry time value with the Deferred CPM Message:
   a. In case of a Pager Mode CPM Standalone Message,
      - If the SIP MESSAGE request contains an Expires header as defined in Appendix C “*CPM-defined Header fields*” [RFC3428], and the value is below the maximum expiry time value per service provider policy, the CPM Participating Function SHALL set the expiry time value to the value contained in the SIP MESSAGE request Expires header field; otherwise the CPM Participating Function SHALL set the expiry time value to the maximum expiry time value per service provider policy.
   b. In case of a Large Message Mode CPM Standalone Message,
      - If the SIP INVITE request contains a Message-Expires header as defined in Appendix C “*CPM-*defined *Header fields*” and the value is below the maximum expiry time value per service

provider policy, the CPM Participating Function SHALL set the expiry time value to the value contained in the SIP INVITE request Message-Expires header field; otherwise the CPM Participating Function SHALL set the expiry time value to the maximum expiry time value per service provider policy.

When a Deferred CPM Message reaches its associated expiry time value, the CPM Participating Function SHALL handle that Deferred CPM Message as described in section 8.3.1.6.8 "*Handling Deferred CPM Messages on Expiry Time*".

8.3.1.6.2 Pushing Deferred CPM Messages

The CPM Participating Function SHALL check if there is a rule in [OMA-XDM-Policy] in which the "enabler" attribute of the '<service>' sub-element inside the '<service-list>' element of the conditions part of the rule is set to "CPM" and the '<action>' element of the rule contains a '<allow-do-not-disturb>' sub-element set to "true".

If no such rule exists, then the CPM Participating Function MAY send an out-of-band notification as described in Appendix I.1.1 "*Out-of-band Deferred CPM Message Notification Format*".

Otherwise, if such a rule does exist, then the CPM Participating Function SHALL wait for the rule to be no longer present, and MAY then send an out-of-band notification as described in Appendix I.1.1 "*Out-of-band Deferred CPM Message Notification Format*".

NOTE: If more than one Deferred CPM Message awaits delivery, the above mentioned out-of-band notifications may only include summary information, as per Appendix I.1.1 "*Out-of-band Deferred CPM Message Notification Format*" and Appendix J.1 "*Deferred Messages Metadata*".

The CPM Participating Function SHALL wait until at least one CPM Client of the CPM User registers. When that happens, and the do-not-disturb flag in the user preferences is not set (determined as per above check in [OMA-XDM-Policy]), the CPM Participating Function SHALL handle each Deferred CPM Message as follows:

1. If the Deferred CPM Message is a Pager Mode CPM Standalone Message, the CPM Participating Function SHALL evaluate which of the registered CPM Clients are suitable to receive the Deferred Pager Mode CPM Standalone Message (see step 8 of section 8.3.1.1 "*Handle a Pager Mode CPM Standalone Message*" for a description of what it means for a CPM client to be suitable) and delivers the message to those CPM Clients as described in section 8.3.1.6.6 "*Sending a Pager Mode Deferred CPM Message*", and removes the message from the Deferred CPM Message queue. If no CPM Client is suitable, the CPM Participating Function keeps the message in the Deferred CPM Message queue.
2. If the Deferred CPM Message is a Large Message Mode CPM Standalone Message, the CPM Participating Function SHALL evaluate which of the registered CPM Clients are suitable to receive the Deferred Large Message Mode CPM Standalone Message (see step 8 of section 8.3.1.1 "*Handle a Pager Mode CPM Standalone Message*" for a description of what it means for a CPM client to be suitable) and delivers the message to those CPM Clients as described in section 8.3.1.6.7 "*Sending a Large Message Mode Deferred CPM Message*". If no CPM Client is suitable, the CPM Participating Function keeps the message in the Deferred CPM Message queue.

8.3.1.6.3 Sending Notifications and Awaiting CPM Client Action

The CPM Participating Function SHALL check if there is a rule in [OMA-XDM-Policy] in which the "enabler" attribute of the '<service>' sub-element inside the '<service-list>' element of the conditions part of the rule is set to "CPM" and the '<action>' element of the rule contains an '<allow-do-not-disturb>' sub-element set to "true".

If such a rule does exist, then the CPM Participating Function SHALL wait for the rule to be no longer present, and SHALL then send a notification as described below.

When the rule on do-not-disturb does not exist (anymore), the CPM Participating Function SHALL determine which notification method to use based on the status of the CPM Clients of the recipient as follows by executing the following steps for each registered CPM Client:

1. If the user's registered CPM Client is suitable to receive at least one of the Deferred CPM Messages present in the Deferred CPM Message queue (see step 8 of section 8.3.1.1 "*Handle a Pager Mode CPM Standalone Message*" for a description of what it means for a CPM client to be suitable, the CPM Participating Function:
    a. SHALL generate a SIP MESSAGE request according to the rules and procedures of [RFC3428];
    b. SHALL include an Accept-Contact header field with the feature-tag set to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg' to indicate that the SIP MESSAGE request is a system message;
    c. SHALL include a P-Asserted-Service header field with the 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg';
    d. SHALL set the Request-URI(s) to the address of this CPM Client;
    e. SHALL include the CPMDeferredMsgMgmtURI as the authenticated originator's CPM Address;
    f. SHALL include the User Agent header to indicate the CPM release version as specified in Appendix D "*Release Version in User-agent and Server headers*";

g. SHALL include information about all Deferred CPM Message(s) in the message body, including the unique message-URI-ID assigned to the message when it was deferred;

h. SHALL include metadata about the Deferred CPM Message(s) as described in Appendix I.1.2 "*In-band Deferred CPM Message Notification Format*";

i. SHALL send the SIP MESSAGE request towards this CPM Client along the signalling path.

If none of the registered CPM Clients is suitable for at least one of the Deferred CPM Messages and if service provider policies allow retrieval of deferred messages (e.g. retrieval might be paused in roaming scenarios), the CPM Participating Function SHALL send an out-of-band notification as described in Appendix I.1.1 "*Out-of-band Deferred CPM Message Notification Format*" to all or selected known devices of the recipient.

NOTE 1: If more than one Deferred CPM Message awaits delivery, the above mentioned out-of-band notifications may only include summary information, as per Appendix I.1.1 "*Out-of-band Deferred CPM Message Notification Format*" and Appendix J.1 "*Deferred Messages Metadata*" , i.e. restrict itself to only send the "number" attribute indicating the number of Deferred Messages currently stored in the CPM Participating Function. For in-band notifications, it may happen that not all metadata about waiting messages will fit into one SIP MESSAGE request. In this case, the CPM Participating Function will send multiple SIP MESSAGE requests.

The CPM Participating Function SHALL take no further action with regards to these Deferred CPM Messages (except for handling them when they reach their associated expiry time value as described in section 8.3.1.6.8 *"Handling Deferred CPM Messages on Expiry Time"*) until one CPM Client takes the initiative to decide what to do with the Deferred CPM Messages.

Upon receiving a SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred' included in the Accept-Contact header targeted to Request-URI CPMDeferredMsgMgmtURI:

1. If the CPM Participating Function determines that there are no Deferred CPM Messages to be managed, it SHALL send a SIP 488 "Not Acceptable Here" response and SHALL include a SIP Warning header with the warning text set to "125 No messages" in the response according to the rules and procedures of [RFC3261].

   Otherwise continue with the rest of the steps;

2. The CPM Participating Function SHALL send a SIP 200 "OK" response, accepting the SDP received;
3. For each Deferred CPM Message identified in the URI list of the SIP INVITE request with the "cpm_action" parameter set to "delete", the CPM Participating Function SHALL delete the Deferred CPM Message from the Deferred CPM Message queue and SHALL send a delivery notification message as described in section 8.2.4.1 "*Sending a Disposition Notification*" to inform the sender of the CPM Message that the message was not delivered if a negative delivery notification was requested by the sender.
4. For each Deferred CPM Message identified in the URI list of the SIP INVITE request with the "cpm_action" parameter set to "interwork", the CPM Participating Function SHALL send the Deferred CPM Message towards the ISF as described in section 6.5 "*Communicating with the ISF and IWF"*.
5. For each Deferred CPM Message identified in the URI-list of the SIP INVITE request with the "cpm_action" parameter set to "store", the CPM Participating Function SHALL store the message in the CPM User's message store as described in [OMA-CPM_TS_MessageStorage] and SHALL send a delivery notification message as described in section 8.2.4.1 "*Sending a Disposition Notification*" to inform the sender of the CPM Message that the message was delivered if a positive delivery notification was requested by the sender.
6. For each Deferred CPM Message identified in the URI list of the SIP INVITE request with the "cpm_action" parameter set to "deliver", or for all Deferred CPM Messages if there is no URI list in the SIP INVITE request at all:
   a. If the deferred message is a Pager Mode CPM Standalone Message, the CPM Participating Function SHALL send the Deferred CPM Message to the CPM Client as described in section 8.3.1.6.6 "*Sending a Pager Mode Deferred CPM Message*".
   b. If the deferred message is a Large Message Mode CPM Standalone Message, the CPM Participating Function SHALL send the Deferred CPM Message to the CPM Client as described in section 8.3.1.6.7 "*Sending a Large Message Mode Deferred CPM Message*".
7. Any Deferred CPM Message not identified in the URI list of the SIP INVITE request is left deferred, and will be added to the next Deferred CPM Message notification to the CPM User. Conversely, handling instructions for any Deferred CPM Message identified in the URI list, but not present in the Deferred CPM Message queue anymore, SHALL be ignored.
8. When receiving a SIP ACK request, the CPM Participating Function SHALL ignore this request.

9. When receiving a SIP BYE request, the CPM Participating Function SHALL process this according to the rules and procedures of [RFC3261].
   NOTE 2: such a SIP ACK or SIP BYE request can arrive at any time after step 2, but must not affect execution of the following steps.
10. If the CPM Participating Function receives another SIP INVITE request from a different client of the same CPM User addressed to the same CPMDeferredMsgMgmtURI and with the same feature tag as indicated above, the CPM Participating Function SHALL handle such overlapping SIP INVITE requests in a first come first served manner.

8.3.1.6.4 Delivering Deferred CPM Messages to the Message Storage Server

The CPM Participating Function SHALL:

1. In case of a Pager Mode CPM Standalone Message,
   a. SHALL remove the Disposition-Notification Header Field in CPIM header of the received CPM Standalone Message as defined in [RFC5438];

      NOTE: If present, the request for read report is not removed by the CPM Participating Function.
   b. If the SIP MESSAGE request contains an Expires header as defined in Appendix C "*CPM-defined Header fields*" [RFC3428], and the value is below the maximum expiry time value per service provider policy, the CPM Participating Function SHALL set the Expires header to the value contained in the SIP MESSAGE request Expires header field; otherwise the CPM Participating Function SHALL set the Expires header to the maximum expiry time value per service provider policy.
2. In case of Large Message Mode CPM Standalone Message,
   a. SHALL remove the Disposition-Notification Header Field in CPIM header of the received CPM Standalone Message as defined in [RFC5438];

      NOTE: If present, the request for read report is not removed by the CPM Participating Function.
   b. If the SIP INVITE request contains a Message-Expires header as defined in Appendix C "*CPM-defined Header fields*" and the value is below the maximum expiry time value per service provider policy, the CPM Participating Function SHALL set the Expires header to the value contained in the SIP INVITE request Message-Expires header field; otherwise the CPM Participating Function SHALL set the Expires header to the maximum expiry time value per service provider policy.
3. SHALL hand over the message to the Message Storage Server according to procedures specified in section 6.3.1 "*Object Store Operation*" of [OMA-CPM_TS_MessageStorage];
4. If the originator of the CPM Standalone Message requested a "positive-delivery" disposition notification as defined in [RFC5438], SHALL generate a "positive-delivery" notification as described in section 8.2.4.1 "*Sending a Disposition Notification*".

   NOTE: If Deferred CPM Messages are handed over to the Message Storage Server as described in this section, advanced functionality of the CPM Participating Function on Deferred CPM Messages (e.g. do-not-disturb or selective notifications with or without delivery to CPM Clients) will not be available.

8.3.1.6.5 Handle a Deferred CPM Message Information Request

NOTE: The Deferred CPM Message information request is handled as an originating network service.

Upon receiving a SIP SUBSCRIBE request with the Event header set to 'deferred-messages', the CPM Participating Function:

1. SHALL perform the actions to verify the authenticated originator's CPM Address and authorize the request and if it is not authorized the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include an SIP Warning header with the warning text set to "127 Service not authorised" in the response according to the rules and procedures of [RFC3261];

   Otherwise, continue with the rest of the steps;
2. SHALL generate a SIP 200 "OK" response to the SIP SUBSCRIBE request according to the rules and procedures of [RFC6665];
3. SHALL set the Contact header of the SIP response to the address of the CPM Participating Function;
4. SHALL include an Expires header field with the value of 0;
5. SHALL send the SIP response towards the CPM Client according to the rules and procedures of the SIP/IP core;

6. SHALL generate an initial SIP NOTIFY request according to the rules and procedures of [RFC6665] with the clarifications in this section;
   i. The CPM Participating Function SHALL set the Subscription-State header field to "terminated" and SHALL include a MIME msginfo+xml body as follows:
      a) SHALL include a <message-list> element;
      b) SHALL include a <message> element for each of the deferred messages:
         1. SHALL include the <message> attribute containing the elements defined in Appendix J "*Deferred Messages event package definition*".
7. SHALL send the NOTIFY request according to the rules and procedures of the SIP/IP core.
8. SHALL terminate the subscription for this event.

8.3.1.6.6 Sending a Pager Mode Deferred CPM Message

When the CPM Participating Function wants to send a Pager Mode Deferred CPM Message, the CPM Participating Function SHALL generate a SIP MESSAGE request according to the rules and procedures of [RFC3428]. The CPM Participating Function:

1. SHALL include only one Accept-Contact header containing the CPM Feature Tag which SHALL be set to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred' percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI*";
2. SHALL set the P-Asserted-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred';
3. SHALL include the authenticated originator's CPM Address in the original SIP MESSAGE request as the authenticated originator's CPM Address of the Pager Mode Deferred CPM Message;
4. SHALL include the Request-URI with the recipient CPM Address;
5. SHALL include a User-Agent header to indicate the OMA CPM release version of the Participating Function as specified in the Appendix D "*Release Version in User-agent and Server headers*";
6. SHALL include the stored information, apart from the ICSI in P-Asserted-Service and Accept-Contact, of the Deferred CPM Message as defined in section 8.3.1.6 "*Defer CPM Standalone Messages*" in the SIP MESSAGE request.
7. SHALL check if the CPM User preferences for history recording <allow-offline-storage> is set to true and if so, it SHALL execute the processing described in 8.5 "*Record CPM Conversation History*";
8. MAY replace each Media Object attached to the CPM Standalone Message with a reference, as described in section 8.3.1.4 "*Replacing Media with a Reference*";
9. If recording is successful, SHALL include the UID information retrieved in section 8.5 "*Record CPM Conversation History"* in the Message-UID header of the SIP MESSAGE request;
10. SHALL send the SIP MESSAGE request according to the rules and procedures of the SIP/IP core.

Upon receiving a SIP 200 "OK" response to the sent SIP MESSAGE request, the CPM Participating Function SHALL remove the Deferred CPM Message from the Deferred CPM Message queue.

If an error response is received during the delivery of the Deferred CPM Message and the Deferred CPM Message was recorded to the Message Storage Server in step 6 above, the CPM Participating Function SHALL remove the message from the CPM User's Message Store as described in [OMA-CPM_TS_MessageStorage] and SHALL keep the message in the Deferred CPM Message queue.

8.3.1.6.7 Sending a Large Message Mode Deferred CPM Message

When the CPM Participating Function wants to send a Deferred Large Message Mode CPM Message, the CPM Participating Function SHALL generate an initial SIP INVITE request according to the rules and procedures of [RFC3261]. The CPM Participating Function:

1. SHALL include an Accept-Contact header with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred' percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI"*;

2. SHALL include a Contact header with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred’ percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *“Coding of the ICSI“*;
3. SHALL set the P-Asserted-Service header field with the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred’;
4. SHALL include the authenticated originator’s CPM Address in the original SIP INVITE request as the authenticated originator’s CPM Address of the Deferred Large Message Mode CPM Message;
5. SHALL include the Request-URI with the recipient CPM Address, as received in the original request;
6. SHALL include a User-Agent header to indicate the OMA CPM release version of the Participating Function as specified in the Appendix D “*Release Version in User-agent and Server headers*”;
7. SHALL include the option tag 'timer' in the Supported header;
8. SHALL copy the Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID (if present) header fields from the stored Large Message Mode CPM Message request, to the generated SIP INVITE request, and if stored, also the values of the recipient-list-history body and Subject;
9. SHALL include the Session-Expires header with the refresher parameter set to "uas" according to the rules and procedures of [RFC4028];
10. SHALL set the delta-seconds value to a value lower than 1800 in the Session-Expires header according to service provider policy and to the rules and procedures of [RFC4028];
11. SHOULD NOT include the Min-SE header field according to the rules and procedures of [RFC4028];
12. SHALL include in the SIP INVITE request a MIME SDP body as a SDP offer according to the rules and procedures of [RFC3264], [RFC4566],[RFC4975] , [RFC6135] and [RFC6714] with the following clarification:
    a. SHALL set the SDP directional media attribute to a=sendonly;
    b. SHALL set the content type as a=accept-types:message/cpim;
    c. SHALL set MSRP URI for the MSRP connection setup as a=path:MSRP URI;
    d. SHALL set the size as a=file-selector:size:actual message size as stored from the original request;
    e. SHALL set the a=setup attribute as “actpass”;
    f. SHALL set the a=accept-types and a=accept-wrapped-types attributes to the same value as the ones from the original request.
13. SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

Upon receiving a SIP 200 “OK” response to the SIP INVITE request, the CPM Participating Function:

1. SHALL start the SIP session timer using the value received in the Session-Expires header according to the rules and procedures of [RFC4028];
2. SHALL act as negotiated, according to [RFC6135], to establish the MSRP connection according to [RFC4975], i.e.:
    a. If the a=setup attribute received in the SDP answer of the 200 “OK” was set to “active” then it SHALL act as a “passive” endpoint and SHALL start listening for the incoming MSRP session;
    b. Else, it SHALL act as an “active” endpoint and SHALL initiate the transport connection for MSRP;
3. SHALL generate and send a SIP ACK request as an acknowledgement of the final response according to the rules and procedures of [RFC3261];
4. MAY replace each Media Object attached to the CPM Standalone Message with a reference, as described in section 8.3.1.4 “*Replacing Media with a Reference*”;
5. SHALL when acting as a “passive” endpoint wait for the MSRP session to be established
6. SHALL generate one or more MSRP SEND requests (depending on whether chunking is used) according to the rules and procedures of [RFC4975] and taking into account the maximum chunk size negotiated according to section 5.2.1, if any, with the following clarifications:
    a. SHALL set To-Path header according to the MSRP URI(s) received in the answer SDP;
    b. SHALL set the content type as Content-Type = message/cpim as described in [RFC3862] and SHALL copy the CPIM cpm.Payload-Type header field as received from the originating leg;
    c. SHALL set “From” header to the authenticated originator’s CPM Address in the original SIP INVITE request ;

7. SHALL send the MSRP SEND request(s) on the established MSRP connection.

Upon receiving a SIP final response different from SIP 200 “OK” to the SIP INVITE request, the CPM Participating Function SHALL generate and send a SIP ACK request as an acknowledgement of the final response according to the rules and procedures of [RFC3261] and keep the message in the deferred storage;

Once the Deferred CPM Message has been successfully transferred via MSRP, the CPM Participating Function:

1. if user preference was set to store the CPM Conversation History, the CPM Participating Function SHALL execute the processing described in section 8.5 “*Record CPM Conversation History*”;
2. SHALL generate and send a SIP BYE request according to the rules and procedures of [RFC3261] and SHALL include the UID information retrieved in section 8.5 “*Record CPM Conversation History”* in the Message-UID header of the SIP BYE request;
3. SHALL release the media resources and remove the Deferred CPM Message from the Deferred CPM Message queue.

If an error response is received on one of the MSRP SEND requests or the TCP connection used for the MSRP session is lost, the CPM Participating Function:

1. SHALL keep the message in the Deferred CPM Message queue;
2. SHALL generate and send a SIP BYE request according to the rules and procedures of [RFC3261];
4. SHALL release the media resources.

8.3.1.6.8 Handling Deferred CPM Messages on Expiry Time

On expiry of a Deferred CPM Message, the CPM Participating Function:

1. SHALL check the user preference <expired> retrieved from XDMS as described in 8.4.1 “*Retrieving User Preferences*”,
2. SHALL determine which procedure to use based on the user preferences setting.
    a. if set to “discard”,
        i. SHALL discard the expired Deferred CPM Message in the Deferred CPM Message queue;
        ii. MAY generate a notification to notify Deferred CPM Message was discarded after expiry.
            1. If at least one of the user’s CPM Clients is registered, the CPM Participating Function:
                a. SHALL generate a SIP MESSAGE request to each registered CPM Client according to the rules and procedures of [RFC3428];
                b. SHALL include an Accept-Contact header field with the feature-tag set to ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg’ to indicate that the SIP MESSAGE request is a system message percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 “Coding of the ICSI“;
                c. SHALL set the P-Asserted-Service header field with the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm. systemmsg’;
                d. SHALL set the Request-URI(s) to registered CPM Address(es) of the intended recipient;
                e. SHALL include the address of the CPM Participating Function in the authenticated originator’s CPM Address;
                f. SHALL include the User Agent header to indicate the CPM release version as specified in Appendix D “*Release Version in User-agent and Server headers*”;
                g. SHALL include information as described in Appendix I.2 “*Notification Format of Deferred CPM Message After Expiry*”
            2. Otherwise, the CPM Participating Function SHALL send an out-of-band notification as described in Appendix I.2 “*Notification Format of Deferred CPM Message After Expiry*”.
        iii. SHALL send a delivery notification message as described in section 8.2.4.1 “*Sending a Disposition Notification*” to inform the sender of the CPM Standalone Message that the message was not delivered if a negative delivery notification was requested by the sender;
    b. if set to “store”,

i. SHALL send the request to store the expired message to the Message Storage Server as described in section 6.3 "*Message and History Operations*" in [OMA-CPM_TS_MessageStorage] and retrieve the information (e.g. UID) of stored headers from Message Storage Server as described in section 6.3 "*Message and History Operations*" in [OMA-CPM_TS_MessageStorage];

ii. MAY generate a notification to notify the Deferred CPM Message was stored to the Message Storage Server after expiry.

   1. If at least one of the user's CPM Clients is registered, the CPM Participating Function:

      a. SHALL generate a SIP MESSAGE request to each registered CPM Client according to the rules and procedures of [RFC3428];

      b. SHALL include an Accept-Contact header field with the feature-tag set to 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg' to indicate that the SIP MESSAGE request is a system message;

      c. SHALL set the Request-URI(s) to registered CPM Address(es) of the intended recipient;

      d. SHALL include the address of the CPM Participating Function in the authenticated originator's CPM Address;

      e. SHALL include the User Agent header to indicate the CPM release version as specified in Appendix D "*Release Version in User-agent and Server headers*";

      f. SHALL include information as described in Appendix I.2 "*Notification Format of Deferred CPM Message After Expiry*".

   2. Otherwise, the CPM Participating Function SHALL send an out-of-band notification as described in Appendix I.2 "*Notification Format of Deferred CPM Message After Expiry*".

iii. SHALL send a delivery notification message as described in section 8.2.4.1 "*Sending a Disposition Notification*" to inform the sender of the CPM Standalone Message that the message was delivered if a positive deliver notification was requested by the sender;

8.3.2 CPM Session Handling

A CPM Participating Function in the terminating network may or may not stay in the media path depending on service provider's policies. These policies may be static such that the CPM Participating Function always stays or always does not stay in the media path, or the policies may depend on parameters such as:

1. If a CPM User has a subscription to the Message Storage Server and a user preference is set to record CPM Conversation History when he receives a CPM Session Invitation;
2. If the CPM Participating Function needs to do charging.
3. If the deferred delivery is enabled for the CPM User and supported by the CPM Participating Function.

When the CPM Participating Function stays in the media path acting as a back-to-back user agent (B2BUA), the following applies:

1. The CPM Participating Function acts as a user agent for both SIP sessions it connects. The B2BUA behaves as a user agent server for the incoming side and as a user agent client for the outgoing side(s).
2. The CPM Participating Function correlates both sides of the SIP session, by mapping the corresponding SIP session identities of the SIP sessions, and it maintains call state for all SIP sessions it handles, thus enabling end-to-end communication.

8.3.2.1 Handle a CPM Session Invitation

Upon receiving a SIP INVITE request of a CPM 1-1 Session or a CPM Group Session, the CPM Participating Function:

1. If the CPM Participating Function requires a specific User Agent version to be supported, the CPM Participating Function SHALL check the "User Agent" header field to determine if the CPM Participating Function supports the User Agent version and if not, the CPM Participating Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header field with the warning text set to "132 Version not supported" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with rest of the steps;

2. If the CPM Participating Function does not allow anonymity and anonymity is requested, the CPM Participating Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header field with the warning text set to “119 Anonymity not allowed” in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
3. MAY check if the Authenticated Originator CPM Address of the SIP INVITE request is included in the chat Blacklist URI-list stored in [OMA-XDM-List]. If it is, then the CPM Participating Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header field with the warning text set to “122 Function not allowed” according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps.
4. MAY check if there is a rule in [OMA-XDM-Policy] in which the “enabler” attribute of the ‘<service>’ sub-element inside the ‘<service-list>’ element of the conditions part of the rule is set to “CPM” and the ‘<action>’ element of the rule contains a ‘<allow-do-not-disturb>’ sub-element set to “true”. If such a rule exists then the CPM Participating Function SHALL return a SIP 480 “Temporarily Unavailable” response according to the rules and procedures of [RFC3261];
5. SHALL check the SIP INVITE request SDP attributes against service provider’s policies specified for the service and for the CPM User. If media types of SDP do not conform to the service provider’s policies, the CPM Participating Function SHALL return a SIP 488 “Not Acceptable Here” response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
6. SHALL check that there is at least one CPM Client registered for the CPM User. If there are no suitable registered CPM Clients for the CPM User (see section 6.6 “*Suitable CPM Clients*” for a description of what it means for a CPM client to be suitable and registered), the CPM Participating Function SHALL check the service provider policy and SHALL follow procedures in the section 8.3.6 “*Delivery Policies in Terminating CPM Participating Function*”. Otherwise, continue with the rest of the steps;
7. SHALL check if in the received SIP INVITE:
   i. the P-Preferred-Service header field is present and carries the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’. In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add P-Asserted-Service header field and set it to ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’; or

   ii. the P-Preferred-Service header field is present and carries the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group’. In this case, the CPM Participating Function SHALL remove the P-Preferred-Service header field and add P-Asserted-Service header field and set it to ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group’; or

   iii. the P-Asserted-Service header field is present and carries the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ or the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group’. If another value is found and/or the CPM Feature tag is not supported by that CPM Participating Function or CPM Controlling Function, the CPM Participating Function MAY reject the SIP INVITE with a SIP 403 “Forbidden” response and a SIP Warning header field with the warning text set to “122 Function not allowed” in the response according to the rules and procedures of [RFC3261] and SHALL not continue with the following steps;
8. If the CPM Participating Function stays in the media path, the CPM Participating Function:
   a. SHALL generate a SIP 200 "OK" response towards the originating leg:
      i. SHALL copy the Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID header fields and values of the received SIP INVITE request into the 200 "OK" response;

      ii. SHALL include a Server header field indicating the CPM release version as specified in Appendix D “*Release Version in User-agent and Server headers (Normative)*”;

      iii. SHALL include a Contact header field with a URI identifying its own address;

      iv. SHALL include a Supported header field with the option tag ‘timer’; and

v. If there was a Session-Expires header field in the original SIP INVITE request, SHALL copy the Session-Expires header field from the original SIP INVITE request according to the rules and procedures of [RFC4028];

vi. If there was no Session-Expires header field in the original SIP INVITE request, SHALL include a Session-Expires header field with the refresher parameter set to "uas" and set the delta-seconds value to a value lower than 1800 in the Session-Expires header according to service provider policy and to the rules and procedures of [RFC4028];

vii. SHALL include the SDP answer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714];

1. SHALL include a media line proposing MSRP media parameters;
2. SHALL supply its own MSRP URI as a=path:MSRP URI;
3. SHALL set the SDP directional media attribute to a=sendrecv;
4. SHALL set the a=setup attribute as "passive";
5. SHALL include an SDP 'msrp-cema' attribute in the MSRP media description of the SDP;

viii. SHALL send the SIP 200 "OK" response towards the incoming leg according to the rules and procedures of SIP/IP core and start listening for incoming MSRP;

b. SHALL determine if the received SIP INVITE request contains a Session-Replaces header field (as defined in Appendix C "*CPM-defined SIP Headers*"):

i. If the header field is present, the CPM Participating Function SHALL forward the SIP INVITE request to all connected CPM Clients of the CPM User on which the CPM Session is being replaced according to rules and procedures of the SIP/IP core and then exit step 8;

ii. Otherwise, if Session-Replaces is not present, it SHALL check the offered Media Streams in the received SIP INVITE request, and determine if any of the CPM User's registered Clients is suitable to receive this request;

1) If no CPM Client is suitable to receive the request, CPM Participating Function SHALL follow procedures in the section 8.3.6 "*Delivery Policies in Terminating CPM Participating Function*" and if none of deferred delivery or interworking applies then it SHALL use the:

a. SIP 480 "Temporarily Unavailable" as error response back. Exit step 8.

c. SHALL behave as a B2BUA according to the rules and procedures of [RFC3261] for the duration of the CPM Session; and

d. SHALL generate the SIP INVITE request(s) according to the rules and procedures of [RFC3261] with the following details:

i. SHALL set the Request-URI to the suitable registered CPM Client's identity, which is either its public GRUU (if supported) or its CPM User's Address;

ii. If public GRUU is not supported, then it SHALL also include a new Accept-Contact header carrying only the +sip.instance parameter and the instance identifier value of the suitable CPM User's registered Client, as well as the tags "explicit" and "require" as described in [RFC3841];

iii. SHALL copy the Accept-Contact header of the incoming SIP INVITE request to the outgoing SIP INVITE request;

iv. SHALL copy the Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID header fields and values of the incoming SIP INVITE request to the outgoing SIP INVITE request;

v. SHALL copy the Contact header field if the 'isfocus' feature parameter as described in Appendix C.1.7 "Contact header field values" is present, otherwise SHALL include a Contact header field with a URI identifying its own address;

vi. other header fields included in the original SIP INVITE SHALL also be populated in the outgoing INVITE, with the following clarifications:

1. SHALL include a Supported header field with the option tag 'timer' and a 'recipient-list-invite' tag,

2. SHALL include a Session-Expires header field with the refresher parameter set to "uas" and set the delta-seconds value to a value lower than 1800 in the Session-Expires header according to service provider policy and to the rules and procedures of [RFC4028], and
3. SHOULD NOT include the Min-SE header field according to the rules and procedures of [RFC4028]; and,
4. a User-Agent header field indicating the CPM release version as specified in Appendix D "*Release Version in User-agent and Server headers (Normative)*";

vii. SHALL copy any recipient-list body received in the incoming SIP INVITE request to the recipient-list-history of the outgoing SIP INVITE request as described in [RFC5364];

viii. SHALL include a SDP body as a SDP offer in the SIP INVITE request based on the received in the SDP as described in as described in section 5.2.1.2 "*SDP Handling at Intermediate Nodes*" and section 5.2.1.4 "*Handling of Media connection parameters for MSRP session matching*". The CPM Participating Function SHALL also set the a=setup attribute in the SDP to the value of "actpass" according to RFC6135;

e. SHALL forward the SIP INVITE request to the SIP/IP core for forking the request to the appropriate user's registered CPM Clients according to the rules and procedures of [RFC3261] and SIP/IP Core.

9. If the CPM Participating Function does not stay in the media path, the CPM Participating Function:
   a. SHALL act as a SIP proxy according to the rules and procedures of [RFC3261] for the duration of the CPM Session;
   b. If the SIP INVITE request contains a Session-Replaces header field (as defined in Appendix C "*CPM-defined Header fields*"), SHALL forward the SIP INVITE request to all connected CPM Clients of the CPM User on which the CPM Session is being replaced according to rules and procedures of the SIP/IP core and then exit step 9.
   c. If the SIP INVITE request does not contain a Session-Replaces header field, SHALL check the offered Media Streams in the request and determine if any of the user's registered CPM Clients is suitable to receive the request.
   d. If at least one CPM Client is suitable to receive the request:
      i. SHALL include in the Reject-Contact header field the instance ID in the +sip.instance feature tag as described in [RFC3841] and [RFC5626] for those CPM Clients that are not suitable to receive the request;
      ii. SHALL set in the Request-Disposition header field the fork-directive attribute as "fork" and the parallel-directive attribute as "parallel".
      iii. SHALL forward the SIP INVITE request to the SIP/IP core which will fork the request to the appropriate user's registered CPM Clients according to the rules and procedures of [RFC3261] and SIP/IP core.
   e. Otherwise, SHALL follow procedures in the section 8.3.6 "*Delivery Policies in Terminating CPM Participating Function*" and if none of deferred delivery or interworking apply then SHALL use the SIP 480 "Temporarily Unavailable" response as the error response back.

If the CPM User rejects the CPM Session invitation, the CPM Client returns a SIP 603 "Decline" response. The CPM Participating Function that has previously accepted the session SHALL:

1. discard any deferred or stored CPM Chat Messages from this CPM Session and
2. terminate the SIP 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.

Upon receiving the first SIP 200 "OK" response from one of those suitable CPM Clients, if the CPM Participating Function is acting as a B2BUA, the CPM Participating Function:

1. SHALL send a SIP ACK request to the CPM Client that has sent a SIP 200 "OK" response, according to the rules and procedures of the [RFC3261] and SIP/IP core; and
2. SHALL check the <actions> element <allow-offline-storage> retrieved from XDMS as described in 8.4.1 "*Retrieving User Preferences*", and if set to "true" or if recording is enabled, it SHALL execute the processing described in 8.5 "*Record CPM Conversation History*" i.e. for the storage of the SIP INVITE request into the Session info Object and for the CPM Group Session also the storage of the Group State Object containing the CPM Group Session Identity and the Participant list).

a. as the CPM Participating Function uses a Session History Folder for the CPM Group Session, the CPM Participating Function SHALL verify if a Session History Folder whose name matches the Contribution-ID exists in the Message Storage Server, and:
   i. If one does exist, SHALL skip the creation of the session info object;
   ii. If one does NOT exist, SHALL first create a Session History Folder for the CPM Group Session with the folder name set to the Contribution-ID value and then create a session info object for this session in this folder and then retrieve a UID from the Message Storage Server for the newly created session info object.

3. SHALL follow the actions indicated in section 8.3.2.1.1 “*Handling of CPM Session responses from CPM Clients*” upon receiving responses from any of the CPM User’s CPM Clients.

8.3.2.1.1 Handling of CPM Session responses from CPM Clients

The CPM Participating Function SHALL handle the responses from the CPM Clients of the CPM User in one of the following ways, subject to service provider policies:

A) The CPM Participating Function MAY establish the MSRP Media Streams as described in section 8.6 “*Media Plane Handling*” with *only the first* CPM Client that has answered with a SIP 200 “OK” response. In this case, the CPM Participating Function receiving a subsequent SIP 200 “OK” response from another CPM Client (on another CPM User’s device) SHALL respond with a SIP ACK to acknowledge the receipt of that response and terminate the CPM Sessions with the subsequent CPM Clients by sending a SIP BYE request according to the procedures described in section 8.3.2.1.2 “*MSRP Media Stream with one CPM Client*”, but SHALL not take any action on establishing an MSRP media session with these CPM Clients.

OR,

B) The CPM Participating Function MAY establish the MSRP Media Streams as described in section 8.6 “*Media Plane Handling*” with *all* the CPM User’s Clients which have answered with 200 “OK” response to the SIP INVITE request, according to the procedures in section 8.3.2.1.3 “*Multiple MSRP Media Streams*”. The CPM Participating Function then checks the Content-type in MSRP SEND request, and until a CPM Chat Message containing either a message from a CPM User, or an “isComposing” indication is received from one of the CPM Clients, the CPM Participating Function continues to forward all incoming CPM Chat Messages to the CPM Clients, as well as any received disposition notifications received from the CPM Clients towards the originating side.

   When a CPM Chat Message containing either a CPM User message (i.e. with user content, not an IMDN or “isComposing”), or an “isComposing” indication is received from a CPM Client, the CPM Participating Function terminates the CPM Sessions with the remaining CPM Clients via a SIP BYE request with the Reason header field populated with the protocol set to SIP and the protocol-cause set to 200 along with an optional protocol-text (e.g. SIP;cause=200;text="Call completed elsewhere").

   OR,

C) The CPM Participating Function MAY establish the MSRP Media Streams as described in section 8.6.“*Media Plane Handling*” with *all* the CPM User’s Clients which have answered with 200 “OK” response to the SIP INVITE request, according to the procedures in section 8.3.2.1.3 “*Multiple MSRP Media Streams*”.

Upon receiving the first SIP 200 "OK" response from one of those appropriate CPM Clients, and when the CPM Participating Function is acting as a SIP proxy, the CPM Participating Function:

1. SHALL forward the SIP 200 "OK" response according to the rules and procedures of [RFC3261] and SIP/IP core;
2. SHALL send a SIP ACK request to the CPM Client that has sent a SIP 200 “OK” response, according to the rules and procedures of the [RFC3261] and SIP/IP core; and
3. SHALL continue to act as a SIP proxy for the duration of the CPM Session.

Upon receiving a SIP final response other than a SIP 200 “OK” response from any of the appropriate CPM Clients, the CPM Participating Function:

1. If a SIP 302 “Temporarily Moved” response was received, it SHALL send the CPM Session Invitation towards the address received in the Contact header of the response according to rules and procedures of [RFC3261];
2. It SHALL generate and send a SIP ACK request to the CPM client from which the response was received

Upon receiving a SIP final response other than a SIP 200 “OK” response from all of the appropriate CPM Clients, the CPM Participating Function:

1. If the SIP INVITE was sent out for a CPM Session, and one of the following SIP responses is received from any receiving CPM Client or other network entities on terminating network:
    A. SIP 480 "Temporarily Unavailable" or,
    B. SIP 408 “Timeout”, or
    C. SIP 487 “Request Terminated”, or
    D. SIP 500 “Server Internal Error”, or
    E. SIP 503 “Service Unavailable”, or
    F. SIP 504 “Server Timeout”,
2. the CPM Participating Function SHALL check if deferred delivery or interworking is enabled, as per the service provider policy about handling undelivered messages, described in section 8.3.6 “*Delivery Policies in Terminating CPM Participating Function*”. If either is enabled, it SHALL generate a SIP 200 "OK" response towards the originating side following the procedures of forwarding a 200 “OK” response above, and SHALL stay in the media path with the originating side.
    i. If deferred delivery is enabled, the CPM Participating Function SHALL store all received messages for later delivery, as described in section 0 “Deferred CPM Session Handling” and if the CPM Session is a Group CPM Session, the CPM Participating Function SHALL also subscribe to the CPM Group Session Participant Information on behalf of the served CPM User;
    ii. Otherwise, if the service provider policy is to deliver the session invitation using another non-CPM service, the CPM Participating Function SHALL send the SIP INVITE request directly to the ISF as described in section 6.5 “*Communicating with the ISF and IWF*”;
3. Otherwise, if the CPM Participating Function acts as a proxy, then it SHALL forward one of the error responses along the signalling path according to [RFC3261].
4. Otherwise, if the CPM Participating Function acts as a B2BUA, then it SHALL generate a SIP response based on one of the error responses and send it along the signalling path according to [RFC3261].

Upon receiving a SIP ACK request, the CPM Participating Function SHALL forward the SIP ACK request to the SIP/IP core according to the rules and procedures of the SIP/IP core for delivery to the first CPM Client that sent a SIP 200 “OK” response.

8.3.2.1.2 MSRP Media Stream with one CPM Client

The CPM Participating Function SHALL follow the procedures described in section 8.6 “*Media Plane Handling*” to set up an MSRP session with the CPM Client that has sent the first SIP 200 “OK” response, according to the rules and procedures of [RFC3261], [RFC4975], [RFC6135] and [RFC6714].

The CPM Participating Function SHALL terminate the remaining dialogs with the other CPM User’s Clients, as follows:

- If a 200 “OK” response has not yet been received from the remaining CPM Clients, the CPM Participating Function SHALL send to the remaining CPM Clients a SIP CANCEL with the Reason header field populated with the protocol set to SIP and the protocol-cause set to 200 along with an optional protocol-text (e.g. SIP;cause=200;text="Call completed elsewhere").
- If another 200 “OK” response is received from another CPM Client of the CPM User, an ACK will be sent to acknowledge the receipt of that response and then the session will be terminated by sending a SIP BYE request, with the Reason header field populated with the protocol set to SIP and the protocol-cause set to 200 along with an optional protocol-text (e.g. SIP;cause=200;text="Call completed elsewhere"). No action is taken to establish the media plane for that session.

8.3.2.1.3 Multiple MSRP Media Streams

For each SIP 200 “OK” response received from any of the CPM Clients CPM User, the CPM Participating Function:

1. SHALL send a SIP ACK request to the CPM Client that has sent a SIP 200 “OK” response, according to the rules and procedures of the [RFC3261];
2. SHALL follow the procedures described in section 8.6 “*Media Plane Handling*” to set up an MSRP session with CPM Client that has sent the SIP 200 “OK” response, according to the rules and procedures of [RFC3261], [RFC4975], [RFC6135] and [RFC6714];

8.3.2.2 Handle a Cancel Request

The CPM Participating Function SHALL handle a received SIP CANCEL request in exactly the same manner as when operating in the originating network, as described in section 8.2.2.2 "*Handling a Cancel Request*".

8.3.2.3 Handle a SIP BYE Request

When a SIP BYE request associated with an existing CPM Session is received by the CPM Participating Function from the originating network (i.e. on the leg towards the CPM Controlling Function, or towards the other CPM Participating Function involved in the CPM session):

1) a CPM Participating Function that acts as a SIP proxy:
    a. SHALL forward the received SIP BYE request along the Signalling Path towards the served CPM User's Client(s) and upon receiving a SIP 200 "OK" response to a SIP BYE request, it SHALL forward the SIP 200 "OK" response along the same Signalling Path from which it has received the SIP BYE request, according to the rules and procedures of [3GPP TS24.229];
2) a CPM Participating Function acts as a B2BUA:
    a. SHALL check if a Reason header field is present in the received SIP BYE request. If a Reason header field is not present (e.g. SIP/IP Core network elements of the originating network did not populate it), then the CPM Participating Function SHALL assume a default value of protocol=SIP;cause=200 value and SHALL continue with step i. below;
    b. If the Reason header field is present, the CPM Participating Function SHALL check the values in the Reason header field to determine how to further handle the SIP BYE:
        i. If the Reason header field (as defined in [RFC3326]) has a value set to SIP;cause=200; and MAY have a text parameter containing an explanatory String (e.g. "Call completed", or other), then the SIP BYE indicates that the other party(ies) in the CPM Session have left;
        ii. The CPM Participating Function MAY store based on service provider policy a Group State Object in the CPM Message Store Server under the CPM Group Session folder where it SHALL populate the latest list of known Participants and if CPM Group Session Data management is supported, it MAY populate also the latest subject, icon and bulletin data for the CPM Group Session Data management, and:
            1. the <status> element with the value set to "removed"; and
            2. the <referred-by> element with the address taken from the Referred-by SIP header field, if present in the SIP BYE;
        iii. The CPM Participating Function MAY store, based on service provider policy, a Conference Information Object in the CPM Message Store Server under the CPM Group Session folder where it SHALL populate the latest list of known Participants and CPM Group Session Data in a conference-info body considering the rule defined in section 9.2.14.3 "*Sending Participant Information Notification*" for the use in a MSRP SEND request:
            1. The CPM Participating Function SHALL include in the <endpoint> element of the <user> element with the "yourown" attribute:
                a) the <status> element including the value "disconnected";
                b) the <disconnection-info> element including:
                    i. SHALL include <when> element including the date and time that the Participant left the CPM Session;
                    ii. SHALL include the <reason> element including the value SIP;cause=410"; and MAY have a text parameter containing an explanatory String (e.g. "Gone", or other);
                c) SHALL include the <by> element with the address taken from the Referred-by SIP header field, if present in the SIP BYE;
        iv. If the Reason header field (as defined in [RFC3326]) has a value set to SIP;cause=480; and MAY have a text parameter containing an explanatory String (e.g. "Bearer unavailable", or other), then the SIP BYE was generated by the CPM Controlling Function as result of expiry of inactivity timer of the CPM Group Session, indicating to the Participant CPM User served by the CPM

Participating Function that the CPM Group Session can be further re-started as CPM Long-lived Group Session.

The CPM Participating Function SHALL store a Group State Object and/or a Conference Information Object in the CPM Message Store Server under the CPM Group Session folder where it SHALL populate the latest list of known Participants and if CPM Group Session Data management is supported, it MAY record also the latest subject, icon and bulletin data for the CPM Group Session Data management;

If the Reason header field has any other values, such as Reason: SIP;cause=503;text="Service Unavailable" as defined in [3GPP TS24.229], then the SIP BYE was generated by a node of the SIP/IP Core and MAY indicate a connection loss with the other side. The CPM Participating Function SHALL store a Group State Object and/or a Conference Information Object in the CPM Message Store Server under the CPM Group Session folder where it SHALL populate the latest list of known Participants and if CPM Group Session Data management is supported, it MAY record also the latest subject, icon and bulletin data for the CPM Group Session Data management;

v. If it receives any further CPM Chat Messages or associated disposition notifications from the served CPM Client(s), due to race conditions until the CPM Client(s) processed the SIP BYE request, it SHALL return a MSRP 481 error response (indicating to the CPM Client that it cannot use the current CPM Session) and SHALL discard the received messages.

c. SHALL generate and send a SIP BYE request to the connected CPM Client(s) of the served CPM User, if any, according to the rules and procedures of [3GPP TS24.229];

d. Upon receiving 200 "OK" SIP response from the served connected CPM User's Clients, and if the action element <allow-offline-storage> is set to true, or recording is enabled, the CPM Participating Function:

   i. SHALL store the CPM Session to the Message Storage Server according to procedures specified in section 8.5.2 "*Record CPM Session*";

e. SHALL generate and send a SIP 200 "OK" response towards the other CPM entity that sent the original SIP BYE request, according to the rules and procedures of the SIP/IP core as described in [3GPP TS24.229];

f. SHALL release the all Media Plane resources associated with the CPM Session.

8.3.2.4 SIP Session Timer Expiry

On expiry of the SIP session timer and if the CPM Participating Function is behaving as a B2BUA for the affected CPM Session, the CPM Participating Function:

1. If the affected CPM Session is being interworked, the CPM Participating Function:
   a. SHALL generate SIP BYE requests for each of the associated IWFs for this SIP session according to the rules and procedures of [RFC3261];
   b. SHALL send these SIP BYE requests according to the rules and procedures of [RFC3261].
2. Otherwise, when the affected CPM Session is not being interworked, the CPM Participating Function:
   a. SHALL generate a SIP BYE request to the CPM Client that missed the SIP session refresh on this CPM Session according to the rules and procedures of [RFC3261];
   b. SHALL send the SIP BYE request to CPM Client on this CPM Session according to the rules and procedures of [RFC3261];
3. If there are no other CPM User's Clients connected in this CPM Session, then the CPM Participating Function:
   a. SHALL generate a SIP BYE request for the other CPM entity of the affected CPM Session according to the rules and procedures of [RFC3261];
   b. SHALL send the SIP BYE request according to the rules and procedures of [RFC3261];
   c. SHALL create and store Group State Object;
4. Upon receiving the 200 OK for the SIP BYE on a leg, it SHALL release all Media Plane resources associated with the CPM Session for that leg, according to the rules and procedures of [RFC3261] and [RFC4028].

If the CPM Participating Function behaves as a SIP proxy, it SHALL act as specified in [RFC4028].

8.3.2.5 Handling a CPM Session Modification Request

The CPM Participating Function SHALL handle a received CPM Session modification request (i.e. a SIP re-INVITE request) in exactly the same manner as when operating in the originating network, as described in section 8.2.2.5 "*Handle a CPM Session Modification Request*".

8.3.2.6 Handling of Participants Information

The CPM Participating Function SHALL handle the Participants Information in exactly the same manner as when operating in the originating network, as described in section 8.2.2.6 "*Handle of Participants Information*".

8.3.2.7 Receive Participant Information Notification

The CPM Participating Function SHALL handle the procedures for receiving Participants Information notifications in exactly the same manner as when operating in the originating network, as described in section 8.2.2.6.1 "*Receive Participant Information Notification*".

8.3.2.8 CPM Group Session Re-join Requests

Not applicable. The rejoin request is considered an originating request and thus is handled by the originating CPM Participating Function, as described in section 8.2.2.7 "*CPM Group Re-join Requests*".

8.3.2.9 Deferred CPM Session handling

Upon receiving a SIP BYE request associated with an existing CPM Session, the CPM Participating Function SHALL proceed as described in sect. 8.2.2.3 "*Handle a SIP BYE Request*".

The CPM Participating Function SHALL stay in the media path of a CPM Session, in view of a deferred delivery to the CPM User of all further CPM Messages exchanged within the CPM Session in the following cases:

1. The SIP BYE was received on the session dialog with the CPM Client with a Reason header field with the reason-protocol set to SIP and a reason-cause set to a value that is different than 200, indicating that the CPM User has not left the session voluntarily;
2. Timeout of the SIP INVITE sent to the CPM user;
3. The MSRP connection towards the CPM Client fails.

For a CPM Group Session, if Participant Information Notifications via Conference Information Object are not supported, the the CPM Participating Function SHALL also subscribe to the focus of the CPM Group Session as described 8.2.2.7.2 "*CPM Long-lived Group Session*", to obtain the latest Participant Information that will be delivered together with the CPM Group Session data during the deferred delivery.

The deferred delivery of a CPM Session is triggered in two ways:

A) CPM Participating Function initiates it when the CPM Client becomes available, i.e. upon receiving 3rd party register;

B) The CPM Client sends a re-join request for a CPM Group session as soon as it comes back online, and the CPM Participating Function receives the re-join before reacting on the 3rd party register and issuing an invitation to the CPM Client. In this case, the CPM Participating Function:

- SHALL not send any SIP INVITE to the CPM Client for the deferred delivery of the CPM Group Session,
- it SHALL use the session setup following the received re-join request also for deferred delivery purposes. The deferred delivery of the missed CPM Group Session data is performed over MSRP.

When disposition notifications are received from the receiving CPM Client, the CPM Participating Function SHALL either send them in a re-started Long-lived CPM Group Session, or as SIP MESSAGE requests if there is no CPM Session is active. In the latter case for a CPM Group Session, the SIP MESSAGE requests SHALL carry the same CPM Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID (if applicable) as the CPM Group Session so that the CPM Client that receives the disposition notifications can associate them with the respective CPM Chat Messages from the CPM Group Session.

The CPM Participating Function SHALL close the delivery session as soon as all stored messages are delivered only if the CPM Session is not active at the time of the delivery.

8.3.2.9.1 Deferred delivery initiated by the CPM Participating Function

8.3.2.9.1.1 The 1-1 CPM Session delivery

If any CPM Sessions data (including associated disposition notifications) is pending deferred delivery for the CPM User, then the CPM Participating Function should process the deferred delivery of the CPM Chat Message(s) and associated disposition notifications grouped per original sender. The order in which the senders are processed may be based on service provider policy (e.g. based on the CPIM DateTime header field).

The CPM Participating Function performing deferred delivery of CPM Chat Message(s) and associated disposition notifications:

1. SHALL generate an initial SIP INVITE request according to rules and procedures of [RFC3261] and with the additional clarification as specified in section 7.3.1.1 *" Initiating a CPM 1-1 Session"*. The SIP INVITE request SHALL have the same Conversation-ID and Contribution-ID as the original CPM Session;
2. SHALL set the Request-URI of the SIP INVITE request to the intended recipient CPM User;
3. SHALL address the request to the CPM User device using the procedures described in section 6.1.2 "*Identifying the recipient device in SIP requests and responses*";
4. SHALL set the P-Asserted-Identity header field to the CPMDeferredMsgMgmtURI value configured as per operator policy identifying the CPM Participating Function;
5. SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 "*Coding of the ICSI*";
6. SHALL include a Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI*" and the device identifier of the sender CPM Client which has sent the original message;
7. SHALL set the P-Asserted-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session';
8. SHALL set the Referred-By header field to the value of the original sender of the CPM Chat Message(s) to be delivered, as specified in section 6.1 "*Authenticated Originator's CPM Address*";
9. SHALL include an SDP body as an SDP offer in the SIP INVITE request based on the received SDP from the original SIP INVITE, as described in section 5.2.1.2 "*SDP Handling at Intermediate Nodes*" and section 5.2.1.4 "*Handling of Media connection parameters for MSRP*", and it SHALL set the SDP directional media attribute to a=sendrecv, indicating that the session is setup for delivery and also accepts disposition notifications to be sent back from the receiving CPM Client;
10. SHALL send the SIP INVITE request towards the CPM User according to rules and procedures of the SIP/IP Core.
11. CPM Participating Function SHALL process the SIP response from the targeted CPM Client according to [RFC3261] with the following clarifications:
    a. Upon receiving a SIP 200 "OK" response, and after responding with the SIP ACK:
        i. SHALL start the SIP Session timer using the value received in the Session-Expires header field according to rules and procedures of [RFC4028];
        ii. SHALL establish the Media Plane and MSRP connection method and destination according to section 5.2.1.4 "Handling of Media connection parameters for MSRP";
        iii. SHALL generate MSRP SEND requests taking into account the maximum chunk size negotiated according to section 5.2.1, if any, and containing a message/CPIM body that includes the content of each deferred CPM Chat Message originated by the CPM User indicated in the Referred-By header field during session setup, as determined in the previous step.
        The following MSRP SEND request shall only be sent when a MSRP 200 OK response has been received to the previous one;
        iv. It SHALL accept the IMDN notifications from the receiving CPM Client, and it SHALL generate corresponding IMDN notifications via SIP MESSAGE towards the sender when the

CPM Session is no longer established with the originating side, as per procedures described in section 5.4.1 "*Generate Delivery Notification*s" and 2 "*Generate Read Report*";

v. once all deferred CPM Chat Messages from the original sender have been delivered (e.g. reception of the MSRP 200 OK of the last MSRP SEND), it SHALL send a SIP BYE request and wait for a 200 "OK" response as described in [RFC3261].

NOTE 1: If at any point during the deferred delivery the CPM Participating Function receives a SIP BYE request, it will respond to the SIP BYE request as described in [RFC3261], and keep any undelivered messages for delivery at another time.

NOTE 2: When a second 200 "OK" response is received on a SIP INVITE request from a CPM User with whom a 1-1 CPM Session is already established, the CPM Participating Function will terminate the existing session by sending a SIP BYE request with Reason header field containing a cause value=200 (Call completed elsewhere). This situation may happen when multiple SIP INVITE requests have been sent to a CPM User, and more than one of these requests are answered with a 200 "OK" response.

b. Upon receiving a SIP 480 "Temporarily Unavailable" error response:

- the CPM Participating Function SHALL stop attempting to deliver any further deferred CPM Chat Message from any sender, until the next call to this procedure.

  NOTE: Based on service provider policy, in addition to the SIP 480, other error codes may also end this procedure.

When the CPM Session is still active, the CPM Participating Function SHALL just proxy any IMDNs returned via MSRP by the receiving CPM Client towards the incoming leg.

If the CPM Session is inactive, the CPM Participating Function SHALL deliver the received IMDNs received via MSRP from the recipient CPM Client for the delivered CPM Chat Message(s), via SIP MESSAGE method.

Based on service provider policy, the CPM Participating Function SHALL deliver the deferred IMDNs in one of the following ways:

1) it MAY send the IMDN in an already on-going MSRP session, if that session is established with the the original sender CPM User, or

2) it MAY send the IMDN via a SIP MESSAGE request as described in section 5.4 "*Disposition Notifications*".

When a delivery notification is received for a deferred CPM Chat Message, the CPM Participating Function is no longer required to keep that message.

8.3.2.9.1.2 CPM Group Session delivery

Upon receiving 3rd party register event notification for a CPM Client, the CPM Participating Function SHALL initiate the deferred delivery of any pending CPM Group Session data (i.e. that the CPM User has missed, or it has incompletely received).

The CPM Participating Function SHALL establish one session per each CPM Group Session delivery, using a SIP INVITE request towards the CPM Client with the Request URI set to the address of the served CPM User, as follows:

a) the SIP INVITE:

i. SHALL contain the same Conversation-ID and Contribution-ID values as the original CPM Group Session, and the latest Participants list known to the CPM Participating Function in the <resource-list> in the body;

ii. the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' included in the Accept-Contact header field and Contact header field, percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 "*Coding of the ICSI*", and all the original CPM Group Session information;

iii. SHALL add in the Contact header field the CPM Group Session Identity (focus URI) of the CPM Group Session with the 'isfocus' feature parameter as described in Appendix C.1.7 "*Contact header field values*" and the sip.instance value of the CPM Client if available;

iv. SHALL set the P-Asserted-Identity header field to the CPM Group Session Identity (focus URI) of the CPM Group Session;

v. SHALL set the P-Asserted-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group';

vi. SHALL include the Referred-By header field in accordance with RFC 3892 [20] in the SIP INVITE request, if the CPM Group Session original SIP INVITE request was generated due to a received SIP REFER request from another conference participant and that received SIP REFER request included a Referred-By header field;

b) the SDP body of the SIP INVITE SHALL be set as per the SDP offer of the original CPM Group Session invitation, as specified in step 7.c.iii, including the SDP directional media attribute, which SHALL be set to a=sendrecv indicating bi-directional media being allowed in the CPM Group Session;

c) The CPM Participating Function SHALL handle the response received from the CPM Client and the SIP ACK as per procedures described in sect. 8.3.2.1 "*Handle a CPM Session Invitation*" for the CPM Participating Function acting as a B2BUA. In addition:

1) Upon receiving SIP "603" response:

i. the CPM Participating Function SHALL discard all deferred CPM Messages and disposition notifications for that CPM Group Session;

ii. the CPM Participating Function SHALL generate a SIP ACK request according to the rules and procedures of of [RFC3261] and send it to the CPM Client

iii. if the CPM Group Session is not active, the CPM Participating Function SHALL

a) SHALL generate a SIP INVITE request according to the rules and procedures of [RFC3261];

b) SHALL set the Request-URI to the address of the target CPM Group Session of CPM Long-lived Group Session;

c) SHALL include a User-Agent header to indicate the OMA CPM release version of the CPM Participating Function as specified in Appendix D "Release Version in User-agent and Server headers";

d) SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' 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;

e) SHALL include a Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' 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;

f) SHALL set the P-Asserted-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group';

g) SHALL include the CPM Address of the CPM User as the authenticated originator's CPM Address in Contact header field, as described in section 6.1 "Authenticated Originator's CPM Address"

h) SHALL add a Supported header field with the option tag 'timer';

i) SHALL include a Session-Expires header field with the refresher parameter set to "uac";

j) SHALL include the Conversation-ID header field of the requested CPM Group Session;

k) SHALL include the Contribution-ID header field of the requested CPM Group Session;

l) SHALL set the delta-seconds value to a value of 1800 or lower in the Session-Expires header according to a service provider configured value and to the rules and procedures of [RFC4028];

m) SHALL include an SDP body as an SDP offer in the SIP INVITE request based on the received SDP from the originating client, as described in section 5.2.1.2 "SDP Handling at Intermediate Nodes";

n) SHALL send the generated SIP INVITE request towards the CPM Controlling Function.

o) When receiving a SIP non-2XX final response to the SIP INVITE request, the CPM Participating Function SHALL send a SIP ACK request to confirm its reception according to [RFC3261] and stop further processing.

p) When receiving a SIP 2XX Final Response, the CPM Participating Function SHALL send a SIP ACK request and continue with the rest of the steps (i.e. the Media Plane SHALL NOT be established).

iv. the CPM Participating Function:

a) SHALL send a SIP BYE with Reason code set to SIP protocol and cause=200 (e.g. SIP;cause=200;text="Call completed") to the conference focus of the CPM Group Session and release the media resources;

b) SHALL unsubscribe from the conference event package for that CPM Group Session focus.

2) Otherwise, for any other SIP error responses it MAY re-attempt the deferred delivery (e.g. 480 “Temporary Unavailable” from the CPM Client), based on service provider policies.

d) upon receiving a SIP 200 “OK” response from the CPM Client, it SHALL return a SIP ACK and once the MSRP session is established:

i. the CPM Participating Function SHALL deliver in order the deferred CPM Messages and disposition notifications of that CPM Group Session to the CPM Client;

ii. The CPM Participating Function MAY deliver Participants Information and if supported, Group Session Data management updates to the CPM Client. The CPM Participating Function:

a. SHALL send a deferred Conference Information Objects if the CPM Client supports Participant Information Notifications via Conference Information Object, otherwise

b. if the CPM Client supports the MIME type of the Group State Object defined in sect. 5.2.8 “*Group State Object*” of [OMA-CPM_TS_MessageStorage] MAY send a interim Group State Object data in a MSRP SEND, if the CPM Client supports the MIME type and schema of the Group State Object defined in sect. 5.2.8 “*Group State Object*” of [OMA-CPM_TS_MessageStorage] in the MSRP SEND;

iii. if the CPM Group Session is still active at the time of the deferred delivery, then the CPM Participating Function:

1) SHALL join the SIP session set up with the CPM Client with the on-going SIP session with the Conference focus to enable the exchange of live CPM Chat Messages within the CPM Group Session;

2) MAY unsubscribe from the conference event package for the CPM Group Session focus, if such subscription is still active.

iv. if the CPM Group Session is inactive at the time of the deferred delivery, then the CPM Participating Function SHALL re-start the CPM Group Session by generating a SIP INVITE towards the CPM Controlling Function as described in section 8.2.2.7 “*CPM Group Re-join Requests*” if it receives any IMDNs, “isComposing” or CPM User messages in the deferred delivery session.

v. If during the deferred delivery procedure of a CPM Group Session, the CPM Participating Function receives from the CPM Client a SIP BYE request with a Reason header field containing a cause value of “200”, then the CPM Participating Function SHALL proceed as described in as described in sect. 8.2.2.3 “*Handle a SIP BYE Request*” and it SHALL also discard all remaining deferred messages for that CPM Group Session.

8.3.2.9.1.3 Deferred delivery performed upon CPM Client re-join

Following a CPM Group Session that was previously interrupted with the CPM Client (due to e.g. connectivity loss), the CPM Participating Function could receive a re-join SIP INVITE request from the CPM Client, before receiving of, or reacting on, a 3rd party register notification informing of the CPM Client registration in IMS.

In such case, the CPM Participating Function SHALL proceed with the handling of the re-join request and SHALL also perform any deferred delivery of the CPM Group Session data missed, or accumulated in the absence of, the CPM Client.

The CPM Participating Function SHALL provide all the pending CPM Group Session data (i.e. that the CPM User has missed, or it has incompletely received) over MSRP messages.

1) If the CPM Group Session is inactive:
   - the CPM Participating Function SHALL deliver Participant Information to the CPM Client. The CPM Participating Function:
     - o if Participant Information via Conference Information Object is supported by the CPM Client, SHALL deliver Conference Information Objects;
     - o otherwise, if the CPM Client supports the MIME type of the Group State Object defined in section. 5.2.8 "*Group State Object*" of [OMA-CPM_TS_MessageStorage], the last Participants Information SHALL be provided by the CPM Participating Function inside a MSRP SEND using the MIME type and schema of the group state object as defined in sect. 5.2.8 "*Group State Object*" of [OMA-CPM_TS_MessageStorage], and
     - o if the CPM Client supports the MIME type of the Group State Object defined in sect. 5.2.8 "*Group State Object*" of [OMA-CPM_TS_MessageStorage] MAY provide the interim Group State Objects (e.g. also including, when supported, the updates of the Group Session Data management that were missed and which were performed during the CPM Group Session)
   - the CPM Participating Function SHALL proceed with the re-start procedure for the CPM Long-lived Group Session. If the CPM Long-lived Group Session could not be re-started and a SIP error response is received from the CPM Controlling Function, the CPM Participating Function SHALL send a SIP BYE to the CPM Client with a Reason header field populated with a reason-protocol SIP and the appropriate reason-cause value of the SIP error received from the focus (e.g. 404, or 403);
   - if the CPM Client has sent any message(s) to the participants of the CPM Group Session, and the CPM Group Session could not be re-started with the CPM Controlling Function, then the CPM Participating Function SHALL send a MSRP failure delivery notification to the CPM Client.
2) If the CPM Group Session is active:
   - it SHALL join the session dialog with the CPM Client with the session dialog of the CPM Group Session with the CPM Controlling Function.

8.3.2.9.1.4 Deferred delivery race conditions

In the unlikely case when the CPM Participating Function has sent out the SIP INVITE for the CPM Group Session deferred delivery, but it immediately receives the CPM Client's re-join request for the same CPM Group Session, the CPM Participating Function SHALL:

- if the CPM Group Session is still active:
  - o keep only one session active with the CPM Client B for the same CPM Group Session, by rejecting the re-join that comes in after the SIP INVITE was already sent out by the CPM Participating Function with a SIP response error 491 "Request pending";
- if the CPM Group Session is inactive:
  - o complete the deferred delivery of the pending CPM Group Session data within the deferred delivery session already initiated by the CPM Participating Function, and then in parallel it SHALL forward the CPM Client B re-join request towards a CPM Controlling Function to re-start the CPM Group Session.

8.3.3 CPM File Transfer Handling

8.3.3.1 Handle a CPM File Transfer Initiation

The CPM Participating Function SHALL handle a CPM File Transfer initiation request (i.e. a SIP INVITE request that includes the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer' in the Accept-Contact header) in exactly the same manner as described for CPM Session Invitations in section 8.3.2.1 "*Handle a CPM Session Invitation*", with the clarification that:

a) the CPM Feature Tag expected and checked SHALL be the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer';

b) SHALL handle the SIP session timer parameters defined in [RFC4028], as specified in the section 8.3.1.2 "*Handle a Large Message Mode CPM Standalone Message*".

With respect to the SDP contents of outgoing SIP INVITE requests the CPM Participating Function SHALL follow the rules and procedures of [RFC5547].

8.3.3.2 Handle a CPM File Transfer Closing Request

The CPM Participating Function SHALL handle the closing of a CPM File Transfer session in exactly the same manner as defined for CPM Sessions in section 8.3.2.3 "*Handle a SIP BYE Request*", with the following clarifications:

1. if the action element <allow-offline-storage> is set to true for the terminating CPM User, then follow procedures described in section 8.3.3.2 "*Handle a CPM File Transfer Closing Request*" steps 1 and 2.

8.3.3.3 Handling Deferred CPM File Transfer File(s)

The CPM Participating Function SHALL support deferred delivery of CPM File Transfer file(s) to the terminating CPM Client. The deferred delivery of the CPM File Transfer file(s) occurs in the following cases:

1) original SIP INVITE expired, based on the a timer of the CPM Participating Function local policy,
2) the terminating CPM Client is offline,
3) the originating or terminating CPM Client loses connectivity,
4) due to an error (see Table 3 below).

Response received on terminating CPM Client or Participating Function	Response sent on originating CPM Client or Participating Function	Store the file
480 Temporarily unavailable	200 OK	Y
408 Request Timeout	200 OK	Y
487 Request Terminated	200 OK	Y
500 Server Internal Error	200 OK	Y
503 Service Unavailable	200 OK	Y
504 Server Timeout	200 OK	Y
600 Busy Everywhere	200 OK	Y
Any other response (including 404 Not Found, 603 Decline and 200 OK)	Received response (that is no mapping is done)	N

Table 3: Handling deferred and delivery mapping response

NOTE: The decision of when the CPM Participating Function accepts the initial SIP INVITE and stays in the media path of a CPM File Transfer is out of the scope of this specification. However, it is RECOMMENDED that the CPM Participating Function stay in the media path to provide better user experience.

If the terminating CPM Participating Function is configured to stay in the media path on behalf of the CPM Client, then the following cases SHALL be supported:

1) In the case when the original SIP INVITE expired, based on the a timer in the local policy the CPM Participating Function:
   a) SHALL cancel the SIP INVITE request toward the terminating CPM Client by sending a SIP CANCEL, with reason header, Reason: SIP ;cause=408 ;text="User not responding";
   b) SHALL store the SDP attributes as defined in Section 8.3.3.7 "*Storing CPM File Transfer Deferred Information*";
   c) SHALL return a SIP "200" OK response with an SDP answer containing the file identities of the files as described in [RFC5547];
   d) SHALL execute the Media Plane procedures as described in section 8.3.3.9 "*Media Plane Handling for MSRP Session*" to receive the files
2) In the case when CPM Client was online, but an error as described in the

cpm.Payload-Type	Payload-Type	Not an absolute matching criteria. It is possible that one Content-Type in the payload was supported by one CPM Client of the CPM

		User) e.g. the one the message was originally delivered to), but is not supported by the CPM Client that obtained the CPM Message via synchronization with the Message Store.

3) Table 2, is received after the SIP INVITE was sent, the CPM Participating Function:
   a) SHALL store the SDP attributes as defined in Section 8.3.3.7 "*Storing CPM File Transfer Deferred Information*";
   b) SHALL return a SIP "200" OK response with an SDP answer containing the file identities of the files as described in [RFC5547];
   c) SHALL execute the Media Plane procedures as described in section 8.3.3.9 "*Media Plane Handling for MSRP Session*" to receive the files
4) In the case when CPM Client is offline, the CPM Participating Function:
   a) SHALL store the SDP attributes as defined Section 8.3.3.7 "*Storing CPM File Transfer Deferred Information*";
   b) SHALL return a SIP "200" OK response with an SDP answer containing the file identities of the files as described in [RFC5547];
   c) SHALL execute the Media Plane procedures as described in section 8.3.3.9 "*Media Plane Handling for MSRP Session*" to receive the files

8.3.3.4 Delivering deferred CPM File Transfer file(s)

CPM Participating Function SHALL support deferred delivery CPM File Transfer file(s) in the following cases:

1) Upon receiving a new SIP INVITE request to resume the transfer of the stored CPM File Transfer file(s), CPM Participating Function SHALL accept the session and establishes an appropriate Media Plane session, according to the procedures described in Section 8.3.3.6 "*Receiving SIP INVITE request for deferred delivery CPM File Transfer file(s)*".
2) SHALL send SIP INVITE as defined in Section 8.3.3.5 "*Generating SIP INVITE request for deferred CPM File Transfer file(s)*" to the terminating CPM Client, upon receiving a 3rd party register of the terminating CPM Client, provided that CPM Participating Function is also serving the terminating CPM Client.
3) The CPM Participating Function MAY send SIP INVITE as defined in Section 8.3.3.5 "*Generating SIP INVITE request for deferred delivery CPM File Transfer file(s)*", periodically without receiving 3rd party registration. The timing and frequency of File Transfer session initiations are subject to Service Provider policies.
4) If the original CPM File Transfer was initiated within a CPM Session, the deferred delivery of the CPM Session will be followed by the deferred delivery of the associated CPM File Transfer(s).

Upon successfully delivery of the deferred CPM File Transfer file(s) to the terminating CPM Client(s), the CPM Participating Function SHOULD remove the deferred CPM File Transfer file(s).

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

For deferred CPM File Transfer file(s) to be delivered to the CPM Client, the CPM Participating Function:

1) SHALL generate a SIP INVITE request according to the rules and procedures of [RFC3261] and [RFC5547];
2) SHALL include a User-Agent header to indicate the OMA CPM release version of the CPM Participating Function as specified in Appendix D "Release Version in User-agent and Server headers";
3) SHALL set the P-Asserted-Identity header field to the identity of the original sender;
4) SHALL set the P-Asserted-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer';
5) SHALL set the Referred-By header to the value of the original sender of the CPM File Transfer file(s) to be delivered, as specified in section 6.1 "*Authenticated Originator's CPM Address*"
6) SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer' percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI"*,
7) SHALL include a Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer' percent encoded in a g.3gpp.icsi-ref media feature tag according to [3GPP TS 24.229] section 7.2A.8.2 *"Coding of the ICSI"*,

8) SHALL insert a URI identifying its own address i.e. the configured CPMDeferredMsgMgmtURI, in the Contact header field of the SIP INVITE request;
9) SHALL include an SDP body as the SDP offer in this SIP INVITE request, as defined in [RFC5547] in the SIP INVITE, based on the original CPM File Transfer SDP offer received (e.g. copy the a=file-selector, a=filetransfer-id attributes stored as described in section 8.3.3.7 "*Storing CPM file Transfer Deferred Information*"),.
10) If the CPM Participating Function is aware that the CPM Client is an older version (e.g. CPM v1.0) or does not support IMDN for CPM File Transfer, then the CPM Participating Function MAY forward the SIP INVITE request without the CPIM/IMDN body. In this case, the CPM Participating Function SHALL act as the receiving entity and generate the requested delivery IMDN back to the sender as described in sect. 5.4.1 "*Generate Delivery Notification*".
11) SHALL send the SIP INVITE request according to the rules and procedures of the SIP/IP core.

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

The CPM Participating Function SHALL handle a SIP INVITE for a CPM File Transfer pull as follows:

1) SHALL verify the received 'file-transfer-id' and 'file-name' SDP attributes with previously stored values. If the SDP does not match, the CPM Participating Function SHALL response with a SIP 488 "Not Acceptable Here" response with the correct SDP parameters.
2) If the CPM File Transfer pull is a resume of an interrupted CPM File Transfer and the CPM Participating Function finds a match for the file attributes above, then it SHALL proceed as described in section 7.4.5.2 "*Responding Side* " assuming the role of the original sender;
3) Otherwise, it SHALL return a SIP 200 "OK" response as described in the following steps.

When sending a SI**P** 200 "OK" response, the CPM Participating Function:

A. SHALL generate a SIP 200 "OK" response according to rules and procedures of [RFC3261];
B. SHALL include a Server header to indicate the OMA CPM release version of the CPM Participating Function as specified in Appendix D "Release Version in User-agent and Server headers";
C. SHALL include the SDP answer containing the file identities of the files as described in [RFC5547];
D. SHALL execute the Media Plane procedures as described in section 7.3.9.1 "MSRP-based Media Streams" to receive the files.

8.3.3.7 Storing CPM File Transfer Deferred Information

When storing a deferred CPM File Transfer file(s) for later delivery receiving to CPM Client, the terminating CPM Participating Function:

1) SHALL store all the CPM File Transfer request headers;
2) SHALL the store file-selector attributes according to[RFC5547]: 'name', 'size', 'type', and 'hash';
3) SHALL store file-transfer-id.

8.3.3.8 Timing between originating and terminating Participating Function

When deploying deferred CPM File Transfer, the timing to trigger the deferred and delivery procedure SHOULD take into account whether deferred and delivery is supported on the terminating side, the timer (or time to trigger the error that signals store and forward is required) SHALL be significantly smaller than the timer used on the originating deferred and delivery process. Consequently, the following recommendations SHOULD be followed:

The timer on the originating CPM Participating Function SHOULD be greater than ½ the SIP INVITE timeout period

The timer on the terminating CPM Participating Function (or time to trigger the error) SHOULD be smaller than ¼ the SIP INVITE timeout period.

8.3.3.9 Media Plane Handling for MSRP Session

For MSRP-based Media Streams, the CPM Participating Function SHALL follow the rules and procedures defined in [RFC4975] and [RFC6714] and section 8.6 "Media Plane Handling", with the following clarification:
1) SHALL maintain a MSRP session for each participant who accepted the MSRP session;
2) SHALL act as MSRP client for sending MSRP SEND request according to [RFC6135];
3) SHALL act as an "active" or "passive" endpoint according to [RFC6135], i.e.:
    a) If the CPM Participating Function was the SDP offerer and it has received in the SDP answer an a=setup attribute set to:

        i) "active", then the terminating CPM Participating Function SHALL always be a "passive" endpoint and SHALL start listening for the incoming MSRP session;
        ii) "passive", then it SHALL act as an "active" endpoint and SHALL initiate the transport connection for MSRP;
    b) If the CPM Participating Function was the SDP answerer and it provided an a=setup attribute set to "passive", then it SHALL start listening for the incoming MSRP session.

4) SHALL establish the MSRP connection towards each participant who accepted the MSRP session according to the MSRP connection parameters in the SDP answer received in the 200 "OK" response according to [RFC6135] and section 5.2.1.4 "*Handling of Media connection parameters for MSRP*".

8.3.4 Sending a Disposition Notification

When receiving a SIP MESSAGE request containing a delivery notification, a read report or an interworking notification, the CPM Participating Function SHALL forward the SIP MESSAGE directly to the served CPM User that is the original sender of the CPM Message, according to the rules and procedures of the SIP/IP core.

The CPM Participating Function SHALL proceed as follows:

1. SHALL extract the <message-id> and disposition type from the IMDN in the SIP MESSAGE request;
2. If there are cached message-ids, it SHALL check whether the extracted message-id is the same as one of the cached message-ids:
    a. If the value is the same, the CPM Participating Function SHALL check whether the extracted disposition type is the same as the cached disposition type:
        i. If a disposition of the same type was already sent, the CPM Participating Function SHALL NOT forward the SIP MESSAGE request towards the original sender of the CPM Message;
        ii. If the disposition received is a delivery notification:
            1. and an interworking notification was already sent for the same CPM Message, then the CPM Participating Function SHALL NOT forward the SIP MESSAGE request towards the original sender of the CPM Message;
        iii. If the disposition received is an interworking notification:
            1. If a display notification was already sent for that message, then the CPM Participating Function SHALL NOT forward the SIP MESSAGE request towards the original sender of the CPM Message;
            2. the message being notified is a CPM Chat Message and a "isComposing" was sent in the CPM Session from any of the CPM User's Clients, then it SHALL NOT forward the SIP MESSAGE request towards the original sender of the CPM Chat Message;
        iv. Otherwise, the CPM Participating Function SHALL cache the message-id and the disposition type and SHALL forward SIP MESSAGE request to the original sender of the CPM Message according to the rules and procedures of the SIP/IP core.
    b. Otherwise, the CPM Participating Function SHALL cache the extracted message-id and the disposition type and SHALL forward the SIP MESSAGE request to the original sender of the CPM Message according to the rules and procedures of the SIP/IP core;

    NOTE: How long message-ids are cached is dependent on service provider policy.
3. If there are no cached message-ids, the CPM Participating Function SHALL cache the extracted message-id and the disposition type and SHALL forward SIP MESSAGE request to the original sender of the CPM Message as described in section 8.3.5. "*Delivery of Disposition Notifications*" and according to the rules and procedures of the SIP/IP core.

8.3.5 Delivery of Disposition Notifications

If the receiving CPM Client is not available, the CPM Participating Function SHALL proceed as follows:

A) If the CPM Message Store is not available for the served CPM User, it SHALL defer the disposition notification, following procedures described in sect. 8.3.1.6 "*Defer Standalone CPM Messages*";

B) Otherwise, if the CPM Message Store is available for the served CPM User, it SHALL store the received disposition notifications in the CPM Message Store, and the CPM User's Clients MAY retrieve them from there once they re-gained connectivity.

For network optimization purposes, the aggregation of IMDNs as specified in [RFC5438] MAY be supported by the CPM Participating Function.

If the aggregated IMDNs are supported, a CPM Participating Function MAY aggregate the IMDNs accordingly before delivering them to its own CPM Clients, subject to Service Provider policies.

The CPM Participating Function MAY check the CPM User preferences retrieved from XDMS as described in 8.4.1 "*Retrieving User Preferences*" if there is a rule in [OMA-XDM-Policy] in which the "enabler" attribute of the '<service>' sub-element inside the '<service-list>' element of the conditions part of the rule is set to "CPM" and the '<action>' element of the rule contains an '<allow-offline-storage>' sub-element set to "true". If such a rule exists, or recording is enabled, it SHALL execute the processing described in 8.5 "*Record CPM Conversation History*" to store the disposition notifications that it delivered successfully to at least one of the served CPM User Clients.

If the received disposition notification needs to be stored, the CPM Participating Function SHALL store the disposition notification in the same folder as the original CPM Message (standalone or CPM Chat Message) or CPM File Transfer originally sent by the served CPM User, with which the received disposition notification is associated.

8.3.6 Delivery Policies in Terminating CPM Participating Function

The CPM Participating Function SHALL handle undelivered CPM Messages or CPM File Transfer based on the service provider policy. If the service provider policy is:

1. To apply deferred delivery, the CPM Participating Function SHALL stay in the media path, and:
    a. if applicable to the received CPM request and if not already done, it SHALL establish an MSRP session with the incoming side, to receive and buffer in view of a deferred delivery:
        i. For a Large Mode CPM Standalone Message: the message as described in section 8.3.1.5 "*Establish MSRP Session for Receiving Large Message Mode CPM Standalone Message*"; or
        ii. For a CPM Session: all the received CPM Chat Messages and any IMDN disposition notifications for previous CPM Chat Messages sent by the served CPM User in that CPM Session;
        iii. For Group CPM Sessions: the CPM Chat Messages and any received IMDN disposition notifications;
        iv. For CPM File Transfer: the file(s) and if applicable, any thumbnail(s) received;
    b. SHALL defer the CPM Message, CPM Session or CPM File Transfer delivery as follows:
        i. For CPM Message, defer as per section 8.3.1.6*"Defer CPM Standalone Messages";*
        ii. For CPM Session with its related CPM Chat Messages, and for Group CPM Sessions also the latest conference information received via the SIP NOTIFY requests: defer as per section 0 "*Deferred CPM Session handling*";
        iii. For CPM File Transfer, defer as per section 8.3.3.3 "*Handling Deferred CPM File Transfer File(s)*";

        and wait until conditions are met for deferred delivery to CPM User's suitable clients, i.e. one of:

        iv. if an IMS registration was received for a CPM Client, then initiate delivery as follows:
            1. For a CPM Pager Mode Message it SHALL follow the section 8.3.1.6.6 "*Sending a Pager Mode Deferred CPM Message*";
            2. For a CPM Large Message Mode it SHALL follow the section 8.3.1.6.7 "*Sending a Large Message Mode Deferred CPM Message*";
            3. For a CPM Session, it SHALL follow the section 8.3.2.9.1 "*Deferred delivery initiated by the CPM Participating Function*";

4. For a CPM File Transfer it it SHALL follow the section 8.3.2.9.1 "*Generating SIP INVITE request for deferred delivery of CPM File Transfer file(s)*";

v. if service provider policies require immediate delivery to the CPM Client on the Primary Device, then check if the CPM Client registered in IMS was the one on the Primary Device:

1. If that is the case, it SHALL NOT continue with the remaining steps;
2. Otherwise, if the CPM Client on Primary Device was not registered in IMS, and an interworking event was received indicating the readiness of the CPM Client on the Primary Device to receive SMS, then initiate any pending deliveries via interworked delivery, by sending the SIP request directly to the ISF as described in section 6.5 "*Communicating with the ISF and IWF*".

2. To deliver CPM Messages (including CPM Chat Messages) or CPM File Transfer using a Non-CPM Communication Service, if <allow-interwork> or <allow-deliver-and-interwork> had not already been checked and selected, the CPM Participating Function:

a. SHALL generate a SIP INVITE request, or a SIP MESSAGE request (whichever is applicable), for the ISF according to the rules and procedures of [3GPP TS.24.229];

b. SHALL copy the Accept-Contact header field of the incoming request to the outgoing request.

c. For SIP INVITE request case, it SHALL also copy the Contact header field from the incoming request to the outgoing request and SHALL insert a URI identifying its own address in the Contact header field;

d. SHALL include a User-Agent header field to indicate the OMA CPM release version of the CPM Participating Function, as specified in Appendix D "*Release Version in User-agent and Server headers*";

e. For the SIP INVITE case, it SHALL include the MIME SDP body received in the SIP INVITE request as a SDP offer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714] with the following clarification, the CPM Participating Function:

i. SHALL supply its own MSRP URI as a=path:MSRP URI;

ii. SHALL set the a=setup attribute as "actpass";

f. SHALL send the SIP request directly to the ISF as described in section 6.5 "*Communicating with the ISF and IWF*";

g. Upon receiving the SIP 200 "OK" response to a SIP INVITE it SHALL respond with a SIP ACK and then it SHALL act as an active endpoint and SHALL initiate the transport connection for MSRP.

3. To send back the error response, if deferring is not an option and interworking had already been selected but returned an error response, then the CPM Participating Function SHALL send back the SIP error response or MSRP SEND error response to the sender of the CPM Standalone Message.

NOTE: The behaviour of the CPM Participating Function when none of the three options above have been enabled is considered to be implementation specific and is out of the scope of this specification.

8.3.7 Interworking Results Handling in the CPM Participating Function

When the CPM Participating Function receives a CPM Interworking Event Report:

A) via SIP MESSAGE:

1) SHALL respond with a SIP 200 "OK" response to the CPM IWF;

2) SHALL check the CPM Interworking Event reported in the <cpm-event-iw> element as described in step C).

B) via MSRP SEND request inside an existing CPM Session with the CPM IWF:

1) SHALL respond with a MSRP 200 "OK" response to the CPM IWF;

2) SHALL check the CPM Interworking Event reported in the <*cpm-event-iw*> element as described in step C).

C) SHALL check if the CPM Interworking Event reported in the <*cpm-event-iw*> element includes:

1) <*failed-iw*> event data, then it SHALL extract the CPM User identity from the <cpm-user-address> child element and SHALL identify the message that has failed delivery from the <message-id> element. The CPM Participating Function SHALL check the service provider policy about handling undelivered messages, for a potential deferred

delivery policy, following the procedures in section 8.3.6 "*Delivery Policies in Terminating CPM Participating Function*";

2) <*ready-for-sms*> event data, then the CPM Participation Function determines that the CPM User identified by the authenticated address in the <cpm-user-address> child element, now has the connectivity necessary to receive SMS messages. The CPM Participation Function:

    (1) SHALL check if any deferred CPM Messages or CPM File Transfer requests are pending for the CPM User.

        (a) If none found, it SHALL ignore the event and skip the next steps;

        (b) Else, if any pending delivery is found:

            (i) the CPM Participation Function SHALL wait for the pre-configured amount of time to determine if the CPM User becomes also registered with SIP/IP core;

            (ii) If the CPM User becomes registered with the SIP/IP core, the CPM Participation Function SHALL trigger the deferred delivery of any pending CPM Messages or CPM File Transfer requests via the appropriate CPM delivery method;

            (iii) Otherwise, if there is no SIP/IP core registration for the CPM User, then it SHALL trigger delivery of any pending CPM Messages or CPM File Transfer requests via interworking to SMS, by sending the SIP request directly to the ISF as described in section 6.5 "*Communicating with the ISF and IWF".*

8.4 User Preferences

The CPM User preferences stored in [OMA-XDM-Policy], User Preferences Profiles stored in [OMA-XDM-Policy] and active User Preference Profile indication stored in [OMA-XDM-UPP] can be accessed and managed according to procedures as specified in [OMA-XDM-Core].

8.4.1 Retrieving User Preferences

For retrieving the user preferences corresponding to the CPM User being served, the CPM Participating Function:

1. SHALL access the User Access Policy document specified in [OMA-XDM-Policy], according to procedures specified in [OMA-XDM-Core], to retrieve the user preferences including the user preferences of any User Preferences Profile;
2. When the CPM Participating Function needs to evaluate device-specific preferences, it SHALL access the UPP Directory document specified in [OMA-XDM-UPP], according to procedures in [OMA-XDM-Core], to identify the active User Preferences Profile specified in <active-upp> element for each CPM User's registered device(s).

8.5 Record CPM Conversation History

Any object that is being stored in CPM User's Message Storage Server and the device local storage, SHALL be according to the data format as described in section 5.2 "*Storage Folders and Objects*" in [OMA-CPM_TS_MessageStorage].

Upon receiving a CPM request (i.e. CPM Standalone Messages, CPM Sessions, CPM File Transfer) or legacy messages (i.e. SMS and MMS) that need to be recorded in the CPM Message Store, the CPM Participating Function:

1. SHALL check if the Message Store folder exists for the conversation to which the CPM request belongs, as follows:
    a. If the CPM request is part of a 1-1 conversation (i.e. 1-1 CPM Standalone Message, 1-1 CPM Session, 1-1 File Transfer, SMS or MMS), the CPM Participating Function SHALL check if a folder exists with a name matching the other user's Authenticated Identity (e.g., as received in the P-Asserted-Identity of the received SIP INVITE or SIP MESSAGE, or 200 "OK" SIP response, or from the SMS and MMS signalling).
    b. If the CPM request is a CPM Group Message, a CPM Group Session or a Group File Transfer, then the CPM Participating Function SHALL check for the existence of the folder identified by the Conversation-ID; else,
2. If the conversation folder exists, then CPM Participating Function SHALL proceed with the storing of the CPM request as per the procedures described in the following sub-sections.
3. If the conversation folder does not exist, then CPM Participating Function SHALL create it as follows:

a. If the CPM request is part of a 1-1 conversation (i.e. 1-1 CPM Standalone Message, 1-1 CPM Session, 1-1 File Transfer, SMS or MMS), the CPM Participating Function SHALL create a folder identified by the other user's identity in the 1-1 Conversation. This identity SHALL be determined based on the other user's Authenticated identity received in the signalling. If an MSISDN based address is available in a SIP URI with a "user=phone" parameter, the MSISDN SHALL be used after conversion to a TEL URI. When an alphanumeric URI is used such as a SIP URI, it SHALL be converted to lower case first. The URI parameters SHALL NOT be included in the folder name.

b. If the CPM request is a CPM Group Message, a CPM Group Session or a Group File Transfer, then the CPM Participating Function SHALL create a folder identified by the Conversation-ID header field value from the SIP request. The CPM Participating Function SHALL proceed to store the objects associated with the CPM communication as described in the following sub-sections.

8.5.1 Record CPM Standalone Message

Upon receiving a SIP MESSAGE request, the CPM Participating Function:

A. SHALL request to store the complete message including all the header fields and bodies of the SIP MESSAGE request to Message Storage Server as described in section 5.2.1 "*Message Object*" and following procedures described in section 6.3.1 "*Object Store Operation*" in [OMA-CPM_TS_MessageStorage];

B. SHALL receive the information (e.g. UID) of stored messages from Message Storage as described in section 6.3.1 "*Object Store Operation*" in [OMA-CPM_TS_MessageStorage].

NOTE: The received SIP IMDN messages (delivery, interworking and display notifications) carried in SIP MESSAGE request SHALL be stored in the same conversation folders in the Message Storage Server, as the original CPM Message, CPM Chat Message or CPM File Transfer sent that they are associated with.

Upon receiving a SIP INVITE request, the CPM Participating Function SHALL extract relevant headers of the request, as described in section 5.2.1 "*Message Object*" in [OMA-CPM_TS_MessageStorage].

After receiving an MSRP SEND request, the CPM Participating Function:

i. SHALL wait until all chunks of the message have been received via subsequent MSRP SEND requests and SHALL reassemble the CPM Standalone Message from these chunks;

ii. SHALL send a request to store the CPM Standalone Message including all headers and headers extracted from the SIP INVITE or SIP REFER request as described in section 6.3.1 "*Object Store Operations*" in [OMA-CPM_TS_MessageStorage];

iii. SHALL receive the information (e.g. UID) of stored messages from Message Storage as described in section 6.3.1 "*Object Store Operation*" in [OMA-CPM_TS_MessageStorage].

The CPM Participating Function SHALL populate the value of the CPM Message object UID into the Message-UID header field of:

1) CPM Pager Mode Standalone Message:

   a) the SIP MESSAGE request sent to the recipient CPM Client, when the CPM Participating Function performs the recording on terminating side, or

   b) In the 200 "OK" response to the SIP MESSAGE returned to the originator CPM Client, when the CPM Participating Function performs the recording on originating side.

2) For CPM Large Message Mode Standalone Message:

   a) SIP BYE request sent to the recipient CPM Client, when the CPM Participating Function performs the recording on terminating side, or

   b) In the 200 "OK" response to the SIP BYE request returned to the originator CPM Client, when the CPM Participating Function performs the recording on originating side.

8.5.2 Record CPM Session

The general process of recording a CPM Session is performed by the CPM Participating Function as follows:

A. SHALL extract and store locally the content received through the signaling plane and Media Plane during the CPM Session. Upon receiving new content, it SHALL store:

1) as live recording: immediately after the content/CPM Chat Messages, were either successfully received from the served CPM User or successfully delivered to the served CPM User, as well as the received disposition notifications for the CPM Chat Messages sent out; or,
2) at termination of the CPM Session: all successfully exchanged CPM Chat Messages in that CPM Session and the received disposition notifications for the CPM Chat Messages sent out.

The CPM Participating Function SHALL create messages formatted according to [RFC2822] corresponding to the received or locally stored content and upload these messages to the Message Storage Server, as described in section 6.3.1 "*Object Store Operation*" in [OMA-CPM_TS_MessageStorage].

If the CPM Sessions is:

1. a 1-1 CPM Session, then the CPM Participating Function SHALL store in the 1-1 conversation folder (dedicated for 1-1 communications with the other user), the session info object as described in section 5.2.3 "*Session Info Object*" in [OMA-CPM_TS_MessageStorage], the CPM Chat Messages exchanged in the 1-1 CPM Session and the received associated disposition notifications , as described in 5.2.1 "*Message Object*" and following procedures described in section 6.3.1 "*Object Store Operation*" in [OMA-CPM_TS_MessageStorage].
2. a CPM Group Session, then the CPM Participating Function SHALL check if a session history folder already exists for the CPM Group Session, with the name equal to the Contribution-ID SIP header field value. If such folder does not exist, then it SHALL create one based on the Contribution-ID SIP header field value. The CPM Participating Function SHALL store all Message Store objects related to the CPM Group Session in the session history folder (session info object, message objects, associated disposition notifications sent out by the served CPM User, and group state objects) as described in 6.3 "*Message and History Operations*" in [OMA-CPM_TS_MessageStorage]. The CPM Participating Function SHALL store:
    a. the received recipient-list, or recipient-list-history, from the body of the SIP INVITE into the <invited-participants> element in the body of the Session Info Object;
    b. the CPM Group Session Identity received in either the SIP INVITE, or in the 200 "OK" to the SIP INVITE, from the CPM Controlling Function in the first Group State Object stored;
    c. the Participants Information in the Group State Object, as well as the <subject>, <icon> and <subject> data received via the conference event package notifications;
    d. if the CPM Chat Messages have been delivered to the served CPM User via interworking to the CPM Client on a Primary Device, it SHOULD store the interworking number (if received from ISF), used for the interworked delivery via SMS or MMS in the "iw-number" attribute of the <groupstate> element in the Group State Object.

       NOTE: This allows the CPM Client to provide a better experience to the CPM User in case the CPM Group Session is Long-lived and the CPM User still has only SMS or MMS capability due to access conditions of the CPM Client on the Primary Device.

Upon receiving a SIP INVITE request, the CPM Participating Function:

a) SHALL extract and locally store the headers of the request as described in 5.2.3 "*Session Info Object*" in [OMA-CPM_TS_MessageStorage]. Note that CPM Participating Function MAY extract and locally store the SDP attributes associated with those media streams (i.e. storing of the SDP in CPM Message Store Server is not required);
b) If the CPM Session is live recording, the CPM Participating Function SHALL store the session info object in the CPM Message Storage Server as soon as the CPM Session was accepted by the CPM User on any of his CPM Client(s) and the Media Plane is established;
c) Otherwise, if the CPM session is recorded only at the end of the SIP session, the CPM Session metadata (session info object, group state objects if applicable) and the CPM Chat Messages are stored in the CPM Message Storage Server only at the end of the SIP session corresponding to the CPM Session.

Upon successfully receiving a CPM Chat Message from the served originator CPM Client, or after successfully delivering it to a served recipient CPM Client, the CPM Participating Function:

i. SHALL wait until all chunks of the CPM Chat Message have been received via subsequent MSRP SEND requests and SHALL reassemble the CPM Chat Message from these chunks.

ii. SHALL store the CPM Chat Message in the CPM Message Storage Server. On the terminating side, the CPM Participating Function SHALL store the CPM Chat Messages after successful delivery to the served CPM Client.

Upon successfully receiving a MSRP SEND message with the MIME application/conference-info+xml body to a served recipient CPM Client, then

- if the MSRP SEND with the conference-info body is the first received in the CPM Group Session with the CPM Controlling Function, or
- if conference-info body includes changes of the Participant list, or CPM Group Session meta data (subject, icon, bulletin, user roles)

and

- if the action element includes an '<allow-store>' sub-element set to "true", or
- a sub-element <allow-offline-storage> set to true,

then CPM Participating Function SHALL store the up-to-date Participant Information, <subject>, <icon> and <bulletin> values in a new Conference Information Object in the CPM Message Store Server under the CPM Session folder. This will result in multiple Conference Information Objects, each of which includes a snapshot state along the course of the CPM Session; The Conference Information Object with the latest timestamp includes the last state of the CPM Session.

8.5.3 Record CPM File Transfer

The general process of recording a CPM File Transfer is as follows: the CPM Participating Function SHALL extract and store locally the content received through the signaling plane and Media Plane during the CPM File Transfer. Upon termination of the CPM File Transfer, the CPM Participating SHALL create a message formatted according to [RFC2822] containing all the locally stored data and upload this message to the Message Storage Server.

Upon receiving a SIP INVITE or a SIP REFER request whose method is set to INVITE, the CPM Participating Function

1. SHALL extract and locally store the headers of the request as described in section 5.2.2 "*File Transfer History Object*" in [OMA-CPM_TS_MessageStorage]-. Note that the CPM Participating Function MAY extract and locally store some of the SDP attributes associated with the files transferred (i.e. storing of the SDP is optional).

Upon receiving an MSRP SEND request, the CPM Participating Function:

1. SHALL wait until all chunks of the file have been received via subsequent MSRP SEND requests and SHALL reassemble the file from these chunks, and,
2. SHALL store the file locally.

Upon termination of the CPM File Transfer i.e. receipt of SIP BYE from the sender CPM Client side, the Participating Function:

1. SHALL construct a CPM File Transfer History object containing the CPM File Transfer information, the received files according to section 5.2.2 *"File Transfer History Object"* in [OMA-CPM_TS_MessageStorage], and,
2. SHALL upload the CPM File Transfer History object to the Message Storage Server and receive its UID as described in section 6.3.1 *"Object Store Operation*" in [OMA-CPM_TS_MessageStorage].
3. SHALL populate the value of the CPM File Transfer object UID into:
   a. the Message-UID header field of the SIP BYE request on the terminating side, or

   the Message-UID header field of the 200 "OK" SIP response to the SIP BYE request on the originating
4. MAY remove the locally stored copy of the file.

8.5.4 Record Interworked Deliveries

After a successful delivery performed by the CPM IWF, the CPM Participating Function SHALL store the CPM Standalone Messages, or CPM Sessions with their CPM Chat Messages and metadata, or CPM File Transfers requests, or any interworked IMDN notifications received from CPM IWF, in the CPM Message Store Server as follows:

A) If the only successful delivery to the CPM User was performed via interworking then:

1. SHALL store the CPM Standalone Message as described in section 8.5.1 "*Record CPM Pager Mode Standalone Message*", with the additional MIME headers described in step C);

2. SHALL store the CPM Sessions information, their CPM Chat Messages and if applicable their Group State Object(s) as described in section 8.5.2 "*Record CPM Session*", with the additional MIME headers described in step C);
3. SHALL store the CPM File Transfer request as described in section 8.5.3 "*Record CPM File Transfer*", with the additional MIME headers described in step C);

B) If a successful delivery was already performed via a CPM delivery to a CPM Client of the CPM User (e.g. on a Secondary Device), before the successful interworked delivery to the CPM Client on a Primary Device, then it SHALL store a message object with:

1. an empty body; and,
2. same MIME headers as the CPM version of the CPM Message, CPM Session or CPM File Transfer, including the original IMDN.message-id header as its value uniquely associates the original CPM message object with this additional legacy message object that indicates the delivery of the CPM Message via non-CPM technologies; and
3. the additional MIME headers described in step C);
4. For CPM Group Sessions:
   a. if it received from ISF the unique interworking number used for the interworked delivery for the messages of that CPM Group Session towards the CPM Client on the Primary Device, then it SHALL store it in the <iw-number> element in a Group State Object, together with the rest of the CPM Group Session data of the GSO;

C) SHALL process the following MIME headers based on the information received from the CPM ISF, or from CPM IWF:

1. SHALL store a Message-Context MIME header for the message object, set to the value of:
   a. the Message-Context SIP header field defined in Appendix C.1.8 and C.2, received in:
      i. the 200 "OK" SIP response to SIP MESSAGE for CPM Pager Mode Standalone Messages, or to the SIP INVITE for CPM Sessions, received from the IWF;
      ii. the 200 "OK" SIP response to the SIP BYE for the CPM Large Message Mode Standalone Message and CPM File Transfer requests;
2. SHALL store a Message-Correlator set as follows:
   a. If Message-Context value was set to"pager-message", indicating SMS delivery:
      i. SHALL calculate the value of the SMS message correlator based on the algorithm defined in Appendix N "*Algorithm for Correlating SMS messages with Objects in the CPM Message Store*";
   b. If Message-Context value was set to"multimedia-message", indicating MMS delivery:
      i. SHALL extract its value from the Message-Correlator SIP header field, defined in Appendix C.1.9 and C.2, as received in the 200 "OK" SIP response to the SIP BYE for the CPM Large Message Mode Standalone Message and CPM File Transfer requests.

8.6 Media Plane Handling

The CPM Participating Function establishes the MSRP Media Streams with a CPM Client of the served CPM User as described in [RFC3261], [RFC4975], [RFC6135] and [RFC6714], with the clarifications specified in the following sub-sections.

8.6.1 Procedures upon Receiving MSRP

Upon receiving an MSRP SEND request the CPM Participating Function:

1. If the MSRP SEND is received from the incoming leg, then it SHALL forward the MSRP SEND request received to the outgoing leg(s), according to the rules and procedures of [RFC4975], [RFC6714] and taking into account the maximum chunk size negotiated according to section 5.2.1 "*SDP Contents for CPM Sessions*", if any.

2. For CPM Sessions:

   A. if the SIP session is setup with more than one of the served CPM User's Clients, and the MSRP SEND request originated from one of the served CPM User's Clients:

i. If service provider policy requires to keep only one of the CPM User’s Clients, i.e. the CPM Client from which it received an indication that is being used by the CPM User for this CPM Session, then:

  a) SHALL check the MSRP message Content-type to see if this is a CPM Chat Message, or an IMDN carrying a display notification, or an “isComposing” notification which indicates that the CPM Client was chosen by the served CPM User to participate in this CPM Session, out of the multiple CPM User’s Clients. If this is the case:

    a) SHALL generate a SIP BYE request to the remaining CPM Clients of the CPM User, with the Reason header field populated with the protocol set to SIP and the protocol-cause set to “200” along with an optional protocol-text (e.g. SIP;cause=200;text="Call completed elsewhere"), according to the rules and procedures of the [RFC3261];

    b) Upon receiving the 200 “OK” response to the SIP BYE from a CPM Client, it SHALL proceed with the release of all Media Plane resources with that CPM Client, associated with the CPM Session, according to the rules and procedures of [RFC3261] and [RFC4028];

    c) SHALL forward the MSRP SEND request towards the terminating leg, as per procedures in section 8.6.2 “*Procedures to Send MSRP”*;

ii. Otherwise, if service provider policy supports direct delivery of originated CPM Chat Messages to all CPM User’s Clients connected to the CPM Session and the message originated from one of the CPM Clients of the served CPM User:

  a) The CPM Participating Function SHALL add the Message-Direction header field as defined in Appendix C.1.9 with the value set to “sent”, to the originating MSRP SEND request that is sent to the other CPM User’s connected Clients using the procedures in section 8.6.2 “Procedures to Send MSRP”.

B. For CPM Group Sessions:

i. when receiving a MSRP SEND request using the Content-Type type application/conference-info+xml and containing the schema Conference Information Object defined in [RFC4575] and if for a CPM Session with a CPM Client a Subscription to Participant Information is active, then the CPM Participating Function:

  a) SHALL NOT forward the MSRP SEND request to this CPM Client, instead the CPM Participating Function SHALL generate a SIP NOTIFY request according to the rules and procedures of [RFC6665] and include a Conference Information Object.

  b) When reporting changes in the Participant information, the CPM Participating Function SHALL expect that CPM Controlling Function SHALL use partial notification, as specified in section 9.2.14.3 “*Sending Participant Information Notifications*” and according to the rules and procedures of [RFC4575];

  c) The CPM Participating Function SHALL identify changes of the status of the CPM Group Session by comparison of the content of the received Conference Information Object with the content of the previous Conference Information Object received in the CPM Group Session.

3. Upon receiving an MSRP SEND request from the incoming leg, which must be delivered to the registered connected CPM Client(s) of the served CPM User, the CPM Participating Function:

A. For each message, SHALL wait if applicable, until all MSRP chunks of the message have been received via subsequent MSRP SEND requests and SHALL reassemble the message from these chunks, for further processing, such as either:

i. re-chunking purposes, if supported and applicable, according to the rules and procedures of [RFC4975]. If chunking negotiation is used, it SHALL consider the maximum MSRP chunk size negotiated as specified in section 5.2.1 1. “*SDP Contents when Initiating or Modifying a CPM Session*”; or

ii. storage purposes, in case the delivery will be successful; or

iii. alternative delivery methods as per service provider policies, in the case of failure to deliver the message;

B. SHALL generate and send one or more MSRP 200 “OK” responses back, one per MSRP SEND request received, according to the rules and procedures of [RFC4975];

C. SHALL forward the MSRP SEND requests to the served CPM Client(s) following the procedures described in section 8.6.2 “*Procedures to Send MSRP”;*

D. Upon receiving the MSRP responses from the served CPM Client(s), it SHALL use the assembled CPM Large Message for either:

i. storage purposes, once the CPM Large Message was successfully delivered to at least one served CPM Client(s) and if the CPM Large Message must be stored in the CPM Message Store (e.g. recording of CPM Large Message is enabled); or

ii. to apply the delivery policies set by the service provider, in case of failure to successfully deliver the complete CPM Large Message to at least one of the served CPM Client(s). In this case, the CPM participating Function SHALL follow the procedures described in section 8.3.6 “*Delivery Policies in Terminating CPM Participating Function*”.

4. SHALL check if there are external contents in the MIME body of the message, and for each Content-Type header set to the value “message/external-body” and whose access-type is set to the value “URL” and whose URL contains the parameter “action=fetch”, the CPM Participating Function:

A. SHALL fetch the Media Object, CPM Standalone Message, CPM File Transfer History, CPM Session History or CPM Conversation History indicated by the URL from the Message Storage Server.

B. SHALL replace the value “message/external-body” in the Content-Type of CPIM body header with an appropriate value depending on the fetched data type.

NOTE: The value for the Content-Type header of the CPIM body can be obtained from the stored data’s Content-Type value (e.g. if stored data is a text file whose Content-Type is “text/plain”, the “text/plain” can be copied to the Content-Type header in the CPM Standalone Message).

C. SHALL include the fetched data in the message/cpim body.

8.6.2 Procedures to Send MSRP

When sending an MSRP SEND request, the CPM Participating Function:

1. SHALL act as an MSRP client for sending MSRP SEND request according to [RFC6135]; and

2. SHALL use the established MSRP connection for the outgoing leg if available, according to the rules and procedures of [RFC4975] and [RFC6714]; or

3. If an MSRP connection with that outgoing leg does not exist yet, it SHALL act as negotiated via the a=setup SDP attribute, as specified in section 8.3.3.9 “*Media Plane Handling for MSRP Session*”; and

a. SHALL establish the MSRP connection according to the MSRP connection parameters in the SDP answer received in the SIP 200 “OK” response:

i. if the SDP answer contained the SDP 'msrp-cema' attribute, SHALL apply the procedures according to [RFC6714];

ii. if the SDP answer did not contain the SDP 'msrp-cema' attribute and:

1. the CPM Participating Function is not configured to apply MSRP interoperability according to section 5.2.1.4 “*Handling of Media connection parameters for MSRP*”, SHALL apply as per [RFC6714] the fall-back procedures to [RFC4975]; else,

2. if the CPM Participating Function is configured to apply MSRP interoperability according to section 5.2.1.4 “*Handling of Media connection parameters for MSRP*”, then it SHALL apply MSRP session matching rules as described in section 5.2.1.4.1 “*Legacy MSRP session matching*”.

4. SHALL forward the MSRP SEND request towards the adjacent MSRP node determined in step 1. or 2. of section 8.6.1 “*Procedures upon Receiving MSRP”,* via the established MSRP connection, as follows:

D. If the CPM Participating Function acts as a MSRP B2BUA, then:

i. SHALL populate in the To-Path header the MSRP URI(s) received in the SDP offer from the originator's CPM Client in accordance with rules and procedures of [RFC6714];

ii. SHALL populate in the From-Path header the CPM Participating Function's own MSRP URI, according to the rules and procedures of [RFC6714];

iii. SHALL select the MSRP connection method for the MSRP SEND request according to section 5.2.1.4. "*Handling of Media connection parameters for MSRP*";

iv. SHALL take into account the maximum chunk size negotiated according to section 5.2.1 "*SDP Contents for CPM Sessions*", if any.

v. If it receives an MSRP response different than 200 "OK" from any of the served CPM User's Clients, the CPM Participating Function SHALL handle the error as specified in [RFC4975]. and SHALL check service provider policies for undelivered messages, as specified in section 8.3.6 "*Delivery Policies in Terminating CPM Participating Function*".

   a) If it MUST disconnect the MSRP session, the CPM Participating Function:

      1) SHALL abort the transmission of further MSRP chunks to that CPM Client;

      2) SHALL proceed to disconnecting that CPM Client, by generating a SIP BYE request according to the rules and procedures of [RFC3261] with Reason code set to SIP protocol and cause=480 (e.g. SIP;cause=480;text="Transmission Failed");

      3) SHALL send the SIP BYE request according to the rules and procedures of SIP/IP core described in [3GPP TS24.229] to this served CPM Client and upon receiving a final response to the SIP BYE request, it SHALL release the media resources associated with this leg;

      4) SHALL determine if that CPM Client resides on a Primary Device, or if that is the only CPM Client of the served CPM User that is connected into the CPM Session, then it:

         a. SHALL apply the service provider policy about handling undelivered messages, as described in section 8.3.6 "*Delivery Policies in Terminating CPM Participating Function*";

vi. Otherwise, if it determines to continue this MSRP session (e.g. to enable a later delivery, or to interwork it to a legacy messaging technology), then it SHALL send a MSRP 200 "OK" response back to the leg from which it has received the MSRP SEND to continue the session.

E. Else, the CPM Participating Function that proxies the MSRP requests can only forward the MSRP SEND as received, to one of the CPM User's registered Clients, and then it SHALL return back the MSRP response received from that CPM Client.

8.6.3 Recording MSRP for CPM Sessions

If it successfully received the originated MSRP SEND requests from a served CPM User's Client, or respectively delivered successfully MSRP SEND requests to at least one CPM Client of the served CPM User, then the CPM Participating Function:

1. If supported, it MAY check the <actions> element <allow-offline-storage> retrieved from XDMS as described in 8.4.1 "*Retrieving User Preferences*":
   a. if is set to "true" it SHALL determine that recording was enabled for the CPM Session,
   b. else, it SHALL NOT continue with the next steps;
2. if recording of the CPM Session is enabled, it SHALL continue with the next step;
3. SHALL execute the procedures described in 8.5 "*Record CPM Conversation History*" to store the MSRP messages for this CPM Session, if the MSRP messages are CPM Chat Messages or received disposition notifications previously requested by the served CPM User.

8.7 Termination of all CPM Sessions of a CPM User due to Administrative Trigger

If the CPM Participating Function supports receiving administrative trigger(s) for removal of a CPM User from all CPM Sessions and CPM Group Sessions, the CPM Participating Function:

1. For each active CPM Session of the CPM User:
    a. SHALL generate and send SIP BYE requests with a Reason header field containing SIP;cause=200;text="Call Completed" according to the rules and procedures of [RFC3261].
    b. SHALL release all the Media Plane resources associated with the released CPM Sessions;
2. In order to remove the CPM user from the inactive CPM Long-lived Group Session(s) in which the CPM User is still a Participant:
    a. SHALL extract for the CPM User, from each CPM Group Session Conversation folder in the CPM Message Store, from the last Group State Object (based on the timestamp) in which the element <removed> is not present under the <status> element:
        i. the related Conversation-ID,
        ii. the related Contribution-ID,
        iii. the related CPM Group Session Identity values; and
        iv. the last list of Participants;
    b. SHALL generate a SIP INVITE request to re-join or re-start each CPM Long-lived Group Session, according to the rules and procedures of [RFC3261] as follows:
        i. SHALL set the Request-URI to the address of the target CPM Group Session Identity extracted from the Group State Object;
        ii. SHALL include an Accept-Contact header field with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session 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;
        iii. SHALL include a Contact header field with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ 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;
        iv. SHALL include a P-Asserted-Service header field with the value of the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group;
        v. SHALL include a P-Asserted-Identity header field with the SIP URI of the CPM User;
        vi. SHALL include the Conversation-ID header field with the value extracted for the CPM Group Session;
        vii. SHALL include the Contribution-ID header field with the value extracted for the CPM Group Session;
        viii. SHALL include the MIME SDP body as a SDP offer as described in section 5.2.1.1 “*SDP Contents when Initiating or Modifying a CPM Session*”.
        ix. SHALL send the SIP INVITE request and process the response according to the rules and procedures of the SIP/IP core as described in [3GPP TS24.229], with the following additions for the SIP responses below;
    c. Upon receiving SIP 403 ”Forbidden“ error response to the SIP INVITE request, the CPM Participating Function SHALL generate and send a SIP ACK request as an acknowledgement of the final response. The CPM Participation Function SHALL repeat steps 2.a and b., for the next CPM Group Session Identity.
    d. Upon receiving SIP 404 “Not Found” error response to the SIP INVITE request, the CPM Participating Function SHALL generate and send a SIP ACK request as an acknowledgement of the final response, and:
        i. SHALL generate a new SIP INVITE request according to the rules and procedures of [RFC3261] for this CPM Group Session, populating the same values of Contribution-ID and Conversation-ID in the respective SIP header fields and using the list of Participants extracted from the Group State Object, as follows:
            1. SHALL set the Request-URI to the service provider’s configured CPM Controlling Function URI;

2. SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session 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;
3. SHALL include a Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' 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;
4. SHALL include a P-Asserted-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group;
5. SHALL include a MIME resource-list body as specified in [RFC5366] with the list of the Participants extracted from the Group State Object;
6. SHALL include a P-Asserted-Identity header field with the SIP URI of the CPM User;
7. SHALL include the MIME SDP body of the requested CPM Group Session as a SDP offer as described in section 5.2.1.1 "*SDP Contents when Initiating or Modifying a CPM Session*";
8. SHALL send the SIP INVITE request and process the response according to the rules and procedures of the SIP/IP core as described in [3GPP TS24.229], with the following additions for the case where it receives a SIP 200 "OK" response when it SHALL continue with step e.;

e. Upon receiving a SIP 200 "OK" response to the SIP INVITE request (for either the re-start request, or the new CPM Group Session request) the CPM Participating Function SHALL handle the response according to the rules and procedures of [3GPP TS24.229], with the following clarifications:
    i. The CPM Participating Function SHALL generate and send a SIP ACK request as an acknowledgement of the final response towards the CPM Controlling Function;
    ii. The CPM Participating Function SHALL initiate the Media Plane as in section 7.3.9 "*Media Plane Handling for CPM Sessions*";

f. SHALL generate a SIP BYE request according to the rules and procedures of [RFC3261], with the Reason header field as defined in [RFC3326] containing a protocol-value set to SIP and a protocol cause=200;

g. For the CPM Long-lived Group Session re-start case, it SHALL set the Request-URI to the CPM Group Session Identity of the CPM Group Session to leave;

h. SHALL send the SIP BYE request according to the rules and procedures of SIP/IP core as described in [3GPP TS24.229];

i. SHALL release all Media Plane resources corresponding to the CPM Group Session being closed.

j. The CPM Participation Function SHALL repeat step 2. for the next CPM Group Session Identity until it completed all the found CPM Group Sessions in the CPM User's Message Store.

Note: The implementation of administration triggers for termination of CPM Sessions for a CPM User (e.g. service provider de-provisioning) is out of the scope of this specification.

Note: If CPM Message Store is not available, the alternative means the CPM Participating Function may use to record relevant CPM Group Session Identity, Contribution-ID Conversation-ID values and participant information are out of scope of this specification.

9. Procedures at CPM Controlling Function

The CPM Controlling Function handles incoming requests in the following manner:

- Upon receiving a Pager Mode CPM Standalone Message (i.e. a SIP MESSAGE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg’ included in the Accept-Contact header field), it SHALL handle this Pager Mode CPM Standalone Message as defined in section 9.1.1 “*Pager Mode CPM Standalone Message Handling*”. If any additional strings are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.
- Upon receiving a Large Message Mode CPM Standalone Message (i.e. a SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg’ included in the Accept-Contact header field), it SHALL handle this Large Message Mode CPM Standalone Message as defined in section 9.1.2 “*Large Message Mode CPM Standalone Message Handling*”. If any additional strings are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.
- When receiving a CPM Session Invitation (i.e. a SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ included in the Accept-Contact header field), it SHALL handle this CPM Session as defined in section 9.2 “*CPM Group Session Handling*”. If any additional strings are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.
- When receiving a CPM File Transfer initiation request (i.e. a SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer’ included in the Accept-Contact header field), it SHALL handle this CPM File Transfer as defined in section 9.3 “*CPM File Transfer Handling*”. If any additional strings are present in the Accept-Contact header field, the CPM Participating Function MAY determine to skip certain CPM services (e.g. Interworking, CPM Message Deferral, storage in CPM Message Store, applicability of blacklists and user preferences, a.o.), subject to service provider policies.

9.1 CPM Standalone Message Handling

9.1.1 Pager Mode CPM Standalone Message Handling

Upon receiving a SIP MESSAGE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg’ included in the Accept-Contact header the CPM Controlling Function:

1. SHALL check if the authenticated originator's CPM Address is an authorized address to use functionalities of the CPM Controlling Function and if not authorized the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “127 Service not authorised” in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
2. If the CPM Controlling Function requires a specific User Agent version to be supported, the CPM Controlling Function SHALL check the User-Agent header field to determine if the CPM Controlling Function supports the User Agent version and if not, the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “132 Version not supported” according to rules and procedures of [RFC3261].

   Otherwise, continue with rest of the steps;
3. If the Request-URI of the SIP MESSAGE request is set to the address of the CPM Controlling Function, the CPM Controlling Function:
   a. SHALL check if the SIP MESSAGE request received for a CPM Ad-hoc Group has anonymity requested and whether anonymity is allowed for the authenticated originator’s CPM Address. If not allowed, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" error response and SHALL include a SIP Warning header with the warning text set to “119 Anonymity not allowed” in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps.

   b. SHALL check if the number of recipients exceeds the maximum allowed by service provider policy. If it does, then the CPM Controlling Function SHALL return a SIP 486 “Busy Here” response and SHALL include a SIP Warning header with the warning text set to “102 Too many recipients” in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   c. SHALL check if the MIME resource-list body includes an empty URI list. If the URI list is empty, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to “129 No destinations” in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   d. SHALL fetch the member list contained in MIME <resource-lists> body according to procedures specified in [RFC5365].

4. Otherwise, the CPM Controlling Function:

   a. SHALL check if the authenticated originator's CPM Address is authorized to send a CPM Standalone Message to the Pre-defined Group by the group policy. If not authorized the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to “127 Service not authorised” in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   b. SHALL check if the CPM Pre-defined Group address in the Request-URI exists in [OMA-XDM-Group]. If the CPM Pre-defined Group address does not exist, then the CPM Controlling Function SHALL return a SIP 404 "Not found" response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   c. SHALL check if the CPM Pre-defined Group address in the Request-URI is specified to support the CPM service by evaluating if the <supported-services> element indicates support for the CPM service. If the CPM Pre-defined Group address does not support the CPM service, then the CPM Controlling Function SHALL return a SIP 488 "Not Acceptable" response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   d. SHALL check if the SIP MESSAGE request received for a CPM Pre-defined Group has anonymity requested and whether privacy is allowed for the authenticated originator’s CPM Address. Allowing privacy for a specific authenticated originator’s CPM Address is defined using <allow-anonymity> element of the CPM Group’s authorization rules as specified in [OMA-XDM-Group]. If not allowed, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to “119 Anonymity not allowed” in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   e. If the CPM Controlling Function requires a specific User Agent version to be supported, the CPM Controlling Function SHALL check the User-Agent header field to determine if the CPM Controlling Function supports the User Agent version and if not, the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “132 Version not supported” in the response according to rules and procedures of [RFC3261].

      Otherwise, continue with rest of the steps;

   f. SHALL check if the Request-URI identifies a CPM Pre-defined Group with empty Group List. If the Group List is empty, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to ”129 No destinations” in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   g. SHALL retrieve the members belonging to the CPM Pre-defined Group by interacting with [OMA-XDM-Group].

5. SHALL send a SIP MESSAGE request towards each CPM Group member, the CPM Controlling Function;

   a. SHALL generate a SIP MESSAGE request according to the rules and procedures of [RFC3428];

  b. SHALL copy the values in Accept-Contact header from the received SIP MESSAGE request in the outgoing SIP MESSAGE request, if any Accept-Contact header was received;
  c. SHALL set the Request-URI to the CPM Address or non-CPM Address of the CPM Group member;
  d. If the received SIP MESSAGE request was sent to a CPM Ad-hoc Group, the CPM Controlling Function SHALL include in the authenticated originator's CPM Address, the address of the sending CPM User;
  e. If the received SIP MESSAGE request was sent to a CPM Pre-defined Group, the CPM Controlling Function SHALL include the authenticated originator’s CPM Address with the following clarification:
     i. SHALL set the authenticated originator’s CPM Address to the CPM Pre-defined Group address;
     ii. If privacy was requested by the sending CPM User, the CPM Controlling Function SHALL include a Referred-By header with anonymous URI;
     iii. Otherwise, SHALL include a Referred-By header with the authenticated originator's CPM Address of the received SIP MESSAGE request, as specified in section 6.1 “*Authenticated Originator’s CPM Address*”;
  f. SHALL include a User-Agent header to indicate the OMA CPM release version of the CPM Controlling Function as specified in Appendix D “*Release Version in User-agent and Server headers*”;
  g. SHALL include all the other received SIP headers as well as the body of the received SIP MESSAGE request in each outgoing SIP MESSAGE request, with the following exceptions:
     - SHALL NOT copy the resource-list body of the received SIP MESSAGE. 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 and SHALL set the Content-Dispostion header field to “recipient-list-history; handling=optional” as described in [RFC5366];
     - the Require header field of the outgoing SIP MESSAGE request SHALL not mention “recipient-list-message”. Instead “recipient-list-message” SHALL be included in the Supported header field;
  h. SHALL send the SIP MESSAGE request towards the SIP/IP core according to the rules and procedures of the SIP/IP core.

6. SHALL send a SIP 202 "Accepted" response along the Signalling Path according to [RFC3261], with the Conversation-ID, Contribution-ID, and if present, the InReplyTo-Contribution-ID header field, copied from the received SIP MESSAGE request into the outgoing SIP 202 “Accepted” response and a Server header as specified in Appendix D “*Release Version in User-agent and Server headers*” SHALL be included.

If an error response is received when attempting to deliver the SIP MESSAGE request to a CPM Address that is one of:

1. 404 “Not Found” or
2. 405 “Method Not Allowed” or
3. 410 “Gone” or
4. 414 “Request URI Too Long” or
5. 415 “Unsupported Media Type” or
6. 416 “Unsupported URI Scheme” or
7. 488 “Not Acceptable Here” or
8. 606 “Not Acceptable”,

the CPM Controlling Function SHALL check service provider policies to determine if interworking is allowed, and if it is, SHALL send the CPM Standalone Message request to the ISF as described in section 6.5 “*Communicating with the ISF and IWF*”.

CPM Standalone Messages with a request for disposition notifications are additionally handled according to section 9.2.15 “*Disposition Notification*”.

9.1.2 Large Message Mode CPM Standalone Message Handling

Upon receiving an initial SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg’ corresponding to Large Message Mode CPM Standalone Message included in the Accept-Contact header, the CPM Controlling Function:

1. SHALL check the authenticated originator's CPM Address and authorize the request, and if not authorized, the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header

with the warning text set to “127 Service not authorised” in the response according to the rules and procedures of [RFC3261];

Otherwise, continue with rest of the steps;

2. If the CPM Controlling Function requires a specific User Agent version to be supported, the CPM Controlling Function SHALL check the User-Agent header field to determine if the CPM Controlling Function supports the User Agent version and if not, the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “132 Version not supported” in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with rest of the steps;

3. If the Request-URI of the SIP INVITE request is set to the address of the CPM Controlling Function, the CPM Controlling Function:

   a. SHALL check if the SIP INVITE request received for a CPM Ad-hoc Group has anonymity requested and whether anonymity is allowed as per the service provider’s policy. If not allowed, the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “119 Anonymity not allowed” in the response according to the rules and procedures of [RFC3261];

      Otherwise, continue with the rest of the steps;

   b. SHALL check from the MIME <resource-lists> body that the maximum number of receivers allowed in a CPM Ad-hoc Group is not exceeded. If exceeded, the CPM Controlling Function SHALL return a SIP 486 “Busy Here” response and SHALL include a SIP Warning header with the warning text set to”102 Too many participants” in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   c. SHALL check if the MIME <resource-lists> body includes an empty URI list. If the URI list is empty, the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “No destinations” in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   d. SHALL fetch the member list contained in MIME <resource-lists> body according to procedures specified in [RFC5366].

4. Otherwise, the CPM Controlling Function:

   a. SHALL check that the CPM Pre-defined Group address in the Request-URI, identifying a CPM Pre-defined Group as defined in [OMA-XDM-Group], is specified to support the CPM service by evaluating if the <supported-services> element indicates support for the CPM service. If the CPM service is not supported then the CPM Controlling Function SHALL return a SIP 404 “Not found” response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   b. SHALL check if anonymity is requested and whether anonymity is allowed for the authenticated originator’s CPM Address. Allowing anonymity for a specific authenticated originator’s CPM Address is defined using <allow-anonymity> element of the CPM Group’s authorization rules as specified in [OMA-XDM-Group]. If not allowed, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to “119 Anonymity not allowed” in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;

5. If the “Session-Expires” header is included with the refresher parameter set to “uac”, the CPM Controlling Function SHALL handle the header field according to procedures of [RFC4028]. Otherwise, the request SHALL be rejected with a SIP 403 “Forbidden” response that will include a SIP Warning header field with the warning text set to “122 Function not allowed”;

6. SHALL behave as a B2BUA according to the rules and procedures of [RFC3261] for the duration of the SIP session;

7. SHALL send a SIP INVITE request towards each CPM Group member as follows:

   a. SHALL generate a SIP INVITE request according to the rules and procedures of [RFC3261];

- b. SHALL copy the values in Contact header, Accept-Contact header, Conversation-ID, Contribution-ID and if applicable the Reply-To, Expires, InReplyTo-Contribution-ID and Subject header fields from the received SIP INVITE request in the outgoing SIP INVITE request;
- c. SHALL include a resource-list-history body in the outgoing SIP INVITE request, containing the addresses for which the “to” and “cc” qualifiers were set in the received SIP INVITE request and SHALL set the Content-Disposition header field to “recipient-list-history; handling=optional” as described in [RFC5366];
- d. SHALL set the Request-URI to the CPM Address or non-CPM Address of the CPM Group member;
- e. SHALL insert in the Contact header the URI identifying its own address, including the "isfocus" feature parameter;
- f. SHOULD include the Session-Expires header with the refresher parameter set to 'uas' according to the rules and procedures of [RFC4028];
- g. SHALL include the Supported header set to "timer";
- h. SHALL include a User-Agent header to indicate the OMA CPM release version of the CPM Controlling Function as specified in Appendix D “*Release Version in User-agent and Server headers*”;
- i. SHALL copy the values of the Privacy header from the received SIP INVITE request received in the outgoing SIP INVITE request, if any Privacy header was received;
- j. SHALL copy the MIME SDP body from the received SIP INVITE request in the outgoing SIP INVITE request as an SDP offer according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714] with the following clarifications:
  - i. SHALL include a media line proposing MSRP media parameters;
  - ii. SHALL supply its own MSRP URI as a=path:MSRP URI;
  - iii. SHALL set the SDP directional media attribute to a=sendonly;
  - iv. SHALL set the size as a=file-selector:size:actual message size.
  - v. SHALL set a=setup attribute as “actpass”;
  - vi. SHALL include an SDP 'msrp-cema' attribute in the MSRP media description of the SDP.
- k. If the received SIP INVITE request was sent to a CPM Ad-hoc Group, the CPM Controlling Function SHALL include the authenticated originator’s CPM Address as specified in section 6.1 “*Authenticated Originator’s CPM Address*” with the following clarifications:
  - i. The URI in the authenticated originator’s CPM Address SHALL be set to the CPM Group Session Identity for this particular CPM Ad-hoc Group assigned by the CPM Controlling Function, according to [RFC5366];
  - ii. SHALL include a Referred-By header with the authenticated originator's CPM Address of the sending CPM User, as specified in section 6.1 “*Authenticated Originator’s CPM Address*”, unless privacy was requested by the sending CPM User, and in that case it SHALL include an anonymous URI in the Referred-By header.
- l. If the received SIP INVITE request was sent to a CPM Pre-defined Group, the CPM Controlling Function SHALL include the authenticated originator’s CPM Address as specified in section 6.1 “*Authenticated Originator’s CPM Address*” with the following clarifications:
  - i. SHALL set the authenticated originator’s CPM Address to the CPM Pre-defined Group address;
  - ii. SHALL include a Referred-By header with the authenticated originator's CPM Address of the received SIP INVITE request, as specified in section 6.1 “*Authenticated Originator’s CPM Address*”, or SHALL include a Referred-By header with anonymous URI, if privacy was requested by the sending CPM User.
- m. SHALL send the SIP INVITE request to SIP/IP core according to the rules and procedures of the SIP/IP core.

Upon receiving the first SIP 200 “OK” response, the CPM Controlling Function:

1. SHALL generate SIP 200 “OK” response for the received SIP INVITE request according to the rules and procedures of [RFC3261] , including:
   - a. the same Conversation-ID, Contribution-ID and if applicable InReplyTo-Contribution-ID, header fields from the received SIP INVITE; and,
   - b. the Require and Supported header fields with the value “timer”; and

   c. a Session-Expires header field with the refresher parameter set to “uac”;

2. SHALL include the SDP received in the response of SIP INVITE request as an answer SDP according to the rules and procedures of [RFC3264], [RFC4566], [RFC4975], [RFC6135] and [RFC6714] with the following clarifications:
   a. SHALL include a media line proposing MSRP media parameters;
   b. SHALL supply its own MSRP URI as a=path:MSRP URI;
   c. SHALL set the SDP directional media attribute to a=recvonly;
   d. SHALL set the a=setup attribute as “passive”;
   e. SHALL include an SDP 'msrp-cema' attribute in the MSRP media description of the SDP;
3. SHALL include a Server header to indicate the OMA CPM release version of the CPM Controlling Function as specified in Appendix D “*Release Version in User-agent and Server headers*”;
4. SHALL include a URI identifying its own address in the Contact header;
5. SHALL send the SIP 200 “OK” response according to the rules and procedures of SIP/IP core and SHALL start listening for incoming MSRP session.

Upon receiving the further SIP 200 “OK” responses and if a SIP ACK request was received from the originating CPM Group member already, the CPM Controlling Function SHALL:

1. generate a SIP ACK request and send it to that CPM Group member according to the rules and procedures of [RFC3261]; and
2. Establish the MSRP session with the CPM Group member that provided the SIP 200 “OK” response as follows:
   a. The CPM Controlling Function SHALL act as an MSRP client for sending MSRP SEND requests according to [RFC6135];
   b. The CPM Controlling Function SHALL act as negotiated in SDP, i.e.:
      i. If it received in the SDP answer an a=setup with a value set to “passive”, then it SHALL act as an "active" endpoint to open the transport connection according to [RFC6135];
      ii. Else, if it received in the SDP answer an a=setup with a value set to “active”, then it SHALL act as a "passive” endpoint and SHALL start listening for incoming for incoming MSRP session, according to [RFC6135];
3. When the MSRP session with the CPM Group member is established, the CPM Controlling Function SHALL duplicate any locally stored MSRP SEND request for the CPM Group member with the following modifications according to the rules and procedures of [RFC4975]:
   a. SHALL supply in the To-Path header the MSRP URI(s) received in the answer SDP from the CPM Group member in accordance with rules and procedures of [RFC6714];
   b. SHALL supply in the From-Path header the CPM Controlling Function’s own MSRP URI, according to the rules and procedures of [RFC6714];
4. SHALL send the MSRP SEND requests towards the CPM Group member via the established MSRP connection taking into account the maximum chunk size negotiated according to section 5.2.1, if any.

When receiving a SIP final response different from SIP 200 “OK” from any of the CPM Group members, the CPM Controlling Function SHALL generate a SIP ACK request and send it to that CPM Group member according to the rules and procedures of [RFC3261].

Upon receiving a SIP CANCEL request, the CPM Controlling Function:

1. SHALL act as UAS to handle the SIP CANCEL request according to the rules and procedures of [RFC3261]; and,
2. SHALL act as UAC to cancel the SIP INVITE request towards each CPM Group Session Participant via the SIP/IP core according to the rules and procedures of [RFC3261].

Upon receiving a SIP ACK request, the CPM Controlling Function:

1. SHALL forward the SIP ACK request towards each CPM Group member that sent a SIP 200 “OK” response according to the rules and procedures of SIP/IP core.
2. SHALL initiate an MSRP session with each CPM Group Member that sent a SIP 200 “OK” response as follows:
   a. The CPM Controlling Function SHALL act as an MSRP client for sending MSRP SEND requests according to [RFC6135];
   b. The CPM Controlling Function SHALL act as negotiated in SDP, i.e.:

i. If it received in the SDP answer an a=setup with a value set to “passive”, then it SHALL act as an "active" endpoint to open the transport connection according to [RFC6135];

ii. Else, if it received in the SDP answer an a=setup with a value set to “active”, then it SHALL act as a "passive” endpoint and SHALL start listening for incoming for incoming MSRP session, according to [RFC6135];

c. The CPM Controlling Function SHALL establish the MSRP connection according to the MSRP connection parameters in the SDP answer received from each CPM Group member in the SIP 200 “OK” respons according to [RFC4975] and [RFC6714].

Upon receiving an MSRP SEND request, the CPM Controlling Function:

1. SHALL generate and send an MSRP 200 OK response to the MSRP SEND request;
2. SHALL duplicate the MSRP SEND request for each CPM Group member with the following modifications according to the rules and procedures of [RFC4975]:

   NOTE: the duplication and subsequent sending may start as soon as all MSRP headers and for the first chunk of the message, the CPIM headers have been received.

   a. SHALL supply in the To-Path header the MSRP URI(s) received in the answer SDP from the CPM Group member in accordance with rules and procedures of [RFC6714];
   b. SHALL supply in the From-Path header the CPM Controlling Function’s own MSRP URI, according to the rules and procedures of [RFC6714].
3. SHALL send the MSRP SEND requests towards each CPM Group member (both CPM Users and/or non-CPM Principals) via the established MSRP connections taking into account the maximum chunk size negotiated according to section 5.2.1, if any;
4. If some CPM Group members have not provided a SIP final response to the SIP INVITE request yet, the CPM Controlling Function SHALL store the received MSRP SEND request locally until a SIP final response has been received from all CPM Group members.

Upon receiving a SIP BYE request from the sending CPM User, the CPM Controlling Function:

1. SHALL respond to the SIP BYE request as described in [RFC3261];
2. SHALL release the resources related to the CPM Group Session towards the sending CPM User; and
3. SHALL send a SIP BYE request towards each CPM Group Participant for which no locally stored content remains to be delivered still; and
4. SHALL release the resources related to the CPM Group Session towards the members towards which a SIP BYE request was send.
5. SHALL when all locally stored content is delivered to a CPM Group Member, send a SIP BYE request towards that CPM Group member and release the resources related to the CPM Group Session with that member

Upon receiving an error response indicating a failure to deliver the SIP INVITE request that is one of

1. 404 “Not Found” or
2. 405 “Method Not Allowed” or
3. 410 “Gone” or
4. 414 “Request URI Too Long” or
5. 415 “Unsupported Media Type” or
6. 416 “Unsupported URI Scheme” or
7. 488 “Not Acceptable Here”or
8. 606 “Not Acceptable”

or there is a failure of an MSRP SEND request to a CPM Address, the CPM Controlling Function SHALL check service provider policies to determine if interworking is allowed, and if it is, SHALL send the SIP INVITE request to the ISF as described in section 6.5 “*Communicating with the ISF and IWF”*.

CPM Standalone Messages with a request for disposition notifications are additionally handled according to section 9.2.15 “*Disposition Notification*”.

9.2 CPM Group Session Handling

Each CPM Group Session is identified by a CPM Group Session Identity, as described in [RFC4579] to identify the conference URI associated with the CPM Group Session.

9.2.1 CPM Group Session Initiation

Upon receiving an initial SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' included in the Accept-Contact header, the CPM Controlling Function SHALL check if the feature tag is supported. If it is not, the CPM Controlling Function SHALL reject the SIP INVITE with a SIP 403 "Forbidden" response and a SIP Warning header field with the warning text set to "122 Function not allowed" in the response according to the rules and procedures of [RFC3261]. Otherwise, the CPM Controlling Function:

1. SHALL check whether the authenticated originator's CPM Address is of a CPM User that is allowed to send the request and if not, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "127 Service not authorized" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with rest of the steps;
2. SHALL check the presence of the "isfocus" feature parameter in the URI of the Contact header and if it is present then the CPM Controlling Function SHALL reject the request with a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "105 isfocus already assigned" in the response according to the rules and procedures of [RFC3261].
3. If the Subject header field is present, it SHALL store it into the <subject> element of the <conference–info> for use as the subject for this CPM Group Session, and SHALL also store the originator Participant authenticated address into the <participant> element of the <subject-ext> extension element;
4. SHALL check if the SDP parameters in the SDP offer are allowed and supported by the CPM Controlling Function according to service provider policy.
   a) If the received SDP parameters are not allowed or the received SDP parameters are allowed but none of the values in the SDP offer are supported (e.g. none of the content types that are offered in the a=accept-wrapper-types attribute are supported), the CPM Controlling Function SHALL return a SIP 488 "Not Acceptable here" with a description of the supported SDP parameters by the CPM Controlling Function;
   b) Otherwise, if the SDP parameters are allowed in a CPM Group Session request, it SHALL check if they are supported in that CPM Group Session:
      i. if the SDP attribute *a=chatroom* defined in [RFC7701] with the CPM reserved chat-token value of '*org.openmobilealliance.groupchat.closed*', indicating a request for a CPM Closed Group Session, is present, (i.e. the SDP attribute value "*a=chatroom:org.openmobilealliance.groupchat.closed"* is present), the CPM Controlling Function SHALL handle the CPM Group Session as per service provider policies:
         1. If service provider policies dictate support for CPM Closed Group Session, then the CPM Controlling Function SHALL handle it as a CPM Closed Group Session and SHALL add the SDP attribute value "*a=chatroom:org.openmobilealliance.groupchat.closed"* to the SIP INVITE request sent to all invited participants;
         2. Otherwise, if the service provider policies dictate support for open CPM Group Session, then the CPM Controlling Function SHALL handle it as an open CPM Group Session and SHALL return a SIP 488 "NOT ACCEPTABLE HERE" error response, with a Warning header field set to "134 Closed group chat not supported" back to originator, according to the rules and procedures of [RFC3261].
5. If the Request-URI of the SIP INVITE request is set to the address of the CPM Controlling Function, the CPM Controlling Function:
   a. SHALL check if the SIP INVITE request received for a CPM Ad-hoc Group has anonymity requested and whether anonymity is allowed by service provider policy. If not allowed, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "119 Anonymity not allowed" in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   b. SHALL check from the MIME resource-lists body that the maximum number of Participants allowed in a CPM Ad-hoc Group is not exceeded. If exceeded, the CPM Controlling Function SHALL return a SIP 486 “Busy Here” response and SHALL include a SIP Warning header with the warning text set to “102 Too many participants” in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   c. SHALL check if the MIME resource-list body includes an empty URI list. If the URI list is empty, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to “129 No destinations” in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   d. SHALL fetch the member list contained in MIME <resource-lists> body according to procedures specified in [RFC5366]. This list is used as the set of CPM Group members that have to be invited to the CPM Group Session;

6. Otherwise, the CPM Controlling Function:

   a. SHALL check that the CPM Pre-defined Group address in the Request-URI, identifying a CPM Pre-defined Group as defined in [OMA-XDM-Group], is specified to support the CPM service by evaluating if the <supported-services> element indicates support for the CPM service. If the CPM service is not supported then the CPM Controlling Function SHALL return a SIP 404 "Not found" response" according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   b. SHALL check if the SIP INVITE request received for a CPM Pre-defined Group has anonymity requested and whether privacy is allowed for the authenticated originator’s CPM Address. Allowing privacy for a specific authenticated originator’s CPM Address is defined using <allow-anonymity> element of the CPM Group’s authorization rules as specified in [OMA-XDM-Group]. If not allowed, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to “Anonymity not allowed” in the response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   c. SHALL determine if the authenticated originator’s CPM Address is allowed the action <allow-initiate-conference> as specified in [OMA-XDM-Group]. If the policy does not allow session initiation or if the requested CPM Feature Tag is not supported by the CPM Controlling Function, then the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “122 Function not allowed” in the response according to the rules and procedures of [RFC3261];

      Otherwise, continue with the rest of the steps.

   d. SHALL validate that the media parameters are acceptable for the CPM Pre-defined Group. If not acceptable, then the CPM Controlling Function SHALL reject the request with a SIP 488 "Not Acceptable Here" response according to the rules and procedures of [RFC3261].

      Otherwise, continue with the rest of the steps;

   e. SHALL check if a CPM Group Session is active for the CPM Pre-defined Group. If there is already an active CPM Group Session, then the CPM Controlling Function SHALL process the request as described in section 9.2.3 “*Participant Joining a CPM Group Session*”.

      Otherwise, continue with the rest of the steps;

   f. SHALL determine if the members of the CPM Pre-defined Group have to be invited to the CPM Group Session by checking if the value of the <invite-members> element is ‘true’. If the value is set to ‘true’, then use the members of the CPM Pre-defined Group as the set of CPM Group members that have to be invited to the CPM Group Session.

7. SHALL use the display name described in section 6.3 “*Display Name*”;

8. SHALL establish a CPM Group Session and allocate a CPM Group Session Identity as described in [RFC4353] and [RFC4579];

9. SHALL invite each of the users listed as the CPM Group members that have to be invited to the CPM Group Session as specified in section 9.2.10 “*Create Session with a Participant*”. If a Session-Replaces header (as defined

in Appendix C "*CPM-defined Header fields*") is included in one of the entries in the MIME resource-lists body (when the CPM Group Session is set up for a CPM Ad-hoc Group), the CPM Controlling Function SHALL ensure that the header is added into the SIP INVITE request sent to that entry.

In the case that the CPM Group did not have an active CPM Group Session yet, the CPM Controlling Function SHALL handle the responses from the invited group members as follows:

i. Upon receiving a SIP final response other than 2xx from an invited CPM Group member, the CPM Controlling Function SHALL remove the invited CPM Group member from the CPM Group Session as specified in section 9.2.11 "*Participant Removing Request*" and generate and send a SIP ACK request to the invited CPM Group member according to the rules and procedures of [RFC3261];

   a. If less than the minimum number of Participants accepted the SIP INVITE, then the CPM Controlling Function SHALL return a SIP 410 "Gone" response to the originator of the CPM Group Session and SHALL skip the remaining steps;

   b. When all invited group members have responded with a SIP final response other than 2xx, the CPM Controlling Function SHALL send an SIP 410 "Gone" error response to the originator of the CPM Group Session and SHALL skip the remaining steps;

ii. Upon receiving the first SIP 200 "OK" response for the SIP INVITE request from one of the invited group members, the CPM Controlling Function:

1. SHALL generate a SIP 200 "OK" response to the SIP INVITE request according to the rules and procedures of [RFC3261] with the following clarifications:
   a. SHALL include Require and Supported headers including the value "timer" and a Session-Expires header with the refresher parameter set to 'uac' and start supervising the SIP Session according to the rules and procedures of [RFC4028];
   b. SHALL include the CPM Group Session Identity with the feature-tag 'isfocus' as described in [RFC4579] and the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' corresponding to CPM Session in the Contact header;
   c. SHALL set the P-Asserted-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group';
   d. SHALL include a User-Agent header to indicate the OMA CPM release version of the CPM Controlling Function as specified in Appendix D "*Release Version in User-agent and Server headers*";
   e. SHALL include the authenticated originator's CPM Address as specified in section 6.1 "*Authenticated Originator's CPM Address*" set to the CPM Controlling Function;
   f. SHALL copy the Conversation-ID, Contribution-ID from the received SIP INVITE request into the outgoing SIP INVITE response. Also Require and Supported headers with the value "timer" will be included;
2. SHALL include in the SIP 200 "OK" response a SDP body as a SDP answer based on the received SDP offer in the SIP INVITE request and local Service Provider policy, as described in section 5.2.1.2 "*SDP Handling at Intermediate Nodes*" and SHALL start listening for incoming MSRP session.
   a. For the case where it handles the session as a CPM Closed Group Session, it SHALL also include the SDP attribute value "a=chatroom:org.openmobilealliance.groupchat.closed"in the SIP response;
   b. If the CPM Controlling Function supports the CPM Group Session Data management , then it SHALL populate in the a=accept-wrapped-types attribute the value of the CPM Group Session Data Content-Type as defined in section 6.8 "*CPM Group Session Data Management*" and the value "multipart/related" in the SDP answer. If the value of the CPM Group Session Data Management Content-Type was included in the a=accept-wrapped-types attribute of the SDP offer, then CPM Group Session Data Management is applicable for the session;
   c. If the CPM Controlling Function supports Participant Information Notifications via Conference Information Object, then it SHALL include in the a=accept-wrapped-types attribute the value "application/conference-info+xml" and the value "multipart/related" (if not already present). If the value application/conference-info+xml was included in the a=accept-wrapped-types attribute of the SDP offer, then Participant Information Notifications via Conference Information Object is applicable for the session;

      NOTE: Service provider policy determines how to handle the sessions when different participants accept different media types;

3. SHALL send a SIP 200 “OK” response along the incoming signalling path;
4. SHALL NOT forward any further responses from the remaining group members that had not responded yet.

iii. SHALL when a SIP ACK request was already received from the initiating CPM user,

1. generate and send a SIP ACK request to the CPM Group member from which the SIP Final response was received; and
2. initiate the Media Plane as defined in section 9.2.12 “Media Plane Handling” if the final response was a SIP 200 “OK” response
3. forward any buffered MSRP messages to the CPM Group member when the Media Plane is established

Upon receiving a SIP ACK request, the CPM Controlling Function:

1. SHALL forward the SIP ACK request towards all Participants that sent a SIP 200 “OK” response.
2. SHALL initiate the Media Plane as defined in section 9.2.12 “*Media Plane Handling*”. As soon as the Media Plane is established within the CPM Group Session, the CPM Controlling Function, subject to service provider policies, MAY buffer MSRP messages exchanged in the CPM Group Session for later delivery to a pending Participant.

9.2.2 Session Cancellation Request

When the CPM Controlling Function receives a SIP CANCEL request from the originator to cancel a CPM Group Session, the CPM Controlling function

1. SHALL act as UAS to handle the SIP CANCEL request according to the rules and procedures of [RFC3261]; and,
2. SHALL if it has not received a SIP final response from any one of the Participants yet, act as UAC to cancel the SIP INVITE request towards each CPM Group member via the SIP/IP core according to the rules and procedures of [RFC3261]. .

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

Upon receiving an initial SIP INVITE request with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ included in the Accept-Contact header, that includes a CPM Pre-defined Group address in the Request-URI, which identifies the CPM Pre-defined Group that is owned by the CPM Controlling Function and it has been determined (as per procedure in section 9.2.1 “*CPM Group Session Initiation*”) that the sender of the SIP INVITE request is joining an already existing CPM Group Session for a CPM Pre-defined Group, the CPM Controlling Function

1. SHALL determine if the authenticated originator’s CPM Address is allowed by the <join-handling> action of the Group’s authorization rules as specified in [OMA-XDM-Group] and if not, then the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “127 Service not authorised” in the response according to the rules and procedures of [RFC3261];

   Otherwise, continue with the rest of the steps*;*
2. SHALL accept the SIP request and generate a SIP 200 "OK" response to the SIP INVITE request as follows:
   a. The CPM Controlling Function SHALL include the stored CPM Conversation Identity, CPM Contribution Identity and, if stored, the CPM Contribution being replied to in the headers defined in Appendix C “*CPM-defined Header fields* ” in the SIP 200 “OK” response;
   b. The CPM Controlling Function SHALL include a SDP body in the SIP 200 "OK" response as a SDP answer as described in section 5.2.1.2 “*SDP Handling at Intermediate Nodes*”;
   c. The CPM Controlling Function SHALL include in the ‘accept types’ SDP media attribute the selected ‘mime types’ supported by the CPM Controlling Function from those contained in the SDP offer in the incoming SIP INVITE request.

Upon receiving a SIP ACK request, the CPM Controlling Function:

1. SHALL initiate the Media Plane as described in section 9.2.12 “*Media Plane Handling*” for this Participant;
2. SHALL send a notification to the Participants that have subscribed to the Participant information, that a CPM User has joined in the CPM Group Session, as specified in section 9.2.14.3 “*Sending Participant Information Notifications*”.

9.2.4 Rejoining CPM Group Session Request

Upon receipt of a SIP INVITE request that includes an Accept-Contact header with the CPM Feature Tag ‘urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session’ and a CPM Group Session Identity in the Request-URI, the CPM Controlling Function:

1. SHALL check whether this CPM Group Session Identity exists for an active CPM Group Session, or for an inactive CPM Group Session that was previously handled by the CPM Controlling Function (i.e. a CPM Long-lived Group Session). If it does not exist, the CPM Controlling Function SHALL return a SIP 404 “Not Found” response and SHALL include a SIP Warning header with the warning text set to “123 Session does not exist” in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
2. SHALL check whether the Conversation-ID in the INVITE request matches the one stored for the CPM Group Session of the existing CPM Group Session Identity, either active or inactive. If it does not match, the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header field with the warning text set to “122 Function not allowed” in the response according to the rules and procedures of [RFC3261].

Otherwise, continue with the rest of the steps;

3. SHALL check CPM Group rules as specified in [OMA-XDM-Group] whether the authenticated originator's CPM Address of the joining CPM User is allowed to join. If it is not allowed to join the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “122 Function not allowed” in the response according to the rules and procedures of [RFC3261].
4. SHALL check the CPM Group Session rules (e.g. metadata and policies) for a Long-lived CPM Group Session and/or for CPM Closed Group Session, whether the authenticated originator's CPM Address of the joining CPM User is allowed to join (by checking the service provider policies for the CPM User, and by checking the last participants list to ensure that the CPM user is among the participants for the case of inactive CPM Long-lived Group Session or for the case of CPM Closed Group Session). If it is not allowed to join the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “122 Function not allowed” in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
5. SHALL, in the case of a CPM Group Session for a CPM Pre-defined Group, check if the <max-participant-count> as specified in [OMA-XDM-Group] is already reached and for a CPM Ad-Hoc Group whether it hasn’t reached the maximum number of participants allowed by the local Service Provider Policy. If reached, the CPM Controlling Function SHALL return a SIP 486 “Busy Here” response and SHALL include a SIP Warning header with the text set to “102 Too many participants” in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
6. check if the SIP INVITE request received for a CPM Ad-hoc Group has anonymity requested and whether anonymity is allowed as per the service provider’s policy. If not allowed, the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “119 Anonymity not allowed” in the response according to the rules and procedures of [RFC3261].
7. SHALL check if the SDP parameters in the SDP offer are allowed and supported by the CPM Controlling Function according to service provider policy. If the received SDP parameters are not allowed or the received SDP parameters are allowed but none of the values in the SDP offer are supported (e.g. none of the content types that are offered in the a=accept-wrapper-types attribute are supported), the CPM Controlling Function SHALL return a SIP 488 “Not Acceptable here” with a description of the supported SDP parameters by the CPM Controlling Function.
8. SHALL use the display name, if a display name is included, according to the priority specified in section 6.3 “*Display Name and Anonymity*”;
9. If the “Session-Expires” header is included with the refresher parameter set to “uac”, the CPM Controlling Function SHALL handle the header field according to procedures of [RFC4028]. Otherwise, the request SHALL be rejected with a SIP 403 “Forbidden” response that will include a SIP Warning header field with the warning text set to “122 Function not allowed”;
10. SHALL accept the SIP request and generate a SIP 200 "OK" response according to the rules and procedures of [RFC3261];
    a. SHALL include the stored CPM Conversation Identity, CPM Contribution Identity and InReplyTo-Contribution-ID in the SIP 200 “OK” response;

   b. SHALL include Require and Supported headers including the value "timer" and a Session-Expires header with the refresher parameter set to 'uac' and start supervising the SIP Session according to the rules and procedures of [RFC4028];
   c. SHALL include in the Contact header field of the SIP 200 "OK" response the CPM group session identity with the feature tag is focus and the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' , 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
   d. SHALL set the P-Asserted-Service header field with the value of the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group';
   e. SHALL include a User-Agent header to indicate the OMA CPM release version of the CPM Controlling Function as specified in Appendix D "Release Version in User-agent and Server headers";

11. SHALL include in the SIP 200 "OK" response a SDP body as a SDP answer based on the received SDP offer in the SIP INVITE request and local Service Provider policy, as described in section 5.2.1.2 "*SDP Handling at Intermediate Nodes*" and SHALL start listening for incoming MSRP session.
12. If the CPM Controlling Function supports the CPM Group Session Data management , then it SHALL populate in the a=accept-wrapped-types attribute the value of the CPM Group Session Data Content-Type as defined in section 6.8 "*CPM Group Session Data Management*" and the value "multipart/related" in the SDP answer in addition to the other supported content types;
13. If the CPM Controlling Function supports Participant Information Notifications via Conference Information Object, then it SHALL include in the a=accept-wrapped-types attribute the value "application/conference-info+xml", and the value "multipart/related" (if not already present);
14. SHALL send the SIP 200 "OK" response according to the rules and procedures of SIP/IP core;
15. If the re-join request re-starts an inactive CPM Long-lived Group Session, the CPM Controlling Function SHALL use the CPM Group Session rules (e.g. metadata and policies) for a Long-lived CPM Group Session and/or for CPM Closed Group Session, to determine which participants have to be invited to the re-started CPM Long-lived Group Session and the type of CPM Group Session, then send out the invitations as per step 8 in sect. 9.2.10 "*Create Session With a Participant* ".
16. SHALL generate and send a SIP NOTIFY request to the CPM Clients and/or CPM Participating Functions, which have subscribed to the Participant Information as specified in section 9.2.14.3 "*Sending Participant Information Notifications*".

Upon receiving a SIP ACK request, the CPM Controlling Function SHALL initiate the Media Plane as specified in section 9.2.12 "*Media Plane Handling*" for this CPM Client.

9.2.5 Adding Participants Request

CPM SIP REFER requests are sent within the scope of the CPM Session dialog to which they are related. Upon receiving a SIP REFER request that includes an Accept-Contact header with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' and the "method" parameter set to "INVITE" in the Refer-To header or there is no "method" parameter, the CPM Controlling Function:

1. SHALL if the Refer-Sub header is not present or is set to "true" in the SIP REFER request, generate and send a SIP 421 "Extension Required" response including a Require header with the option tag "norefersub".

   Note: any network entities along the path which act as a B2BUA SHOULD populate the Refer-Sub header field with a value of "false".
2. In case of CPM Pre-defined Group, SHALL perform actions to verify that the authenticated originator's CPM Address of the inviting CPM User is allowed to invite other users by checking the <allow-invite-users-dynamically> action as specified in [OMA-XDM-Group] and if not authorized, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "127 Service not authorised" in the response according to the rules and procedures of [RFC3261];
3. In case of a CPM Ad-hoc group: it SHALL check the type of CPM Group Session, and if it is a CPM Closed Group Session it SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "127 Service not authorised" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of steps;

4. SHALL check whether privacy is allowed for the authenticated originator's CPM Address, when anonymity is requested with the Privacy header containing the tag 'id'. If not allowed, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "119 Anonymity not allowed" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of steps;
5. SHALL extract from the SIP REFER request the address(es) of the user(s), to be invited, either:
   a. from the Refer-To header field, according to the rules and procedures of [RFC3515] and [RFC7647] when inviting one user; or
   b. from the MIME resource-list body indicated by the Content-ID header that has same value as the CID in the Refer-To header field, according to the rules and procedures of [RFC5368] for inviting multiple users.
6. SHALL generate a SIP 200 "OK" final response to the SIP REFER request according to the rules and procedures of [RFC7647]
7. SHALL include in the response to the SIP REFER request a Refer-Sub header set to "false" according to the rules and procedures of [RFC4488];
8. SHALL include in the response to the SIP REFER request a Supported header with the option tag "norefersub" according to the rules and procedures of [RFC4488];
9. SHALL send the response according to the rules and procedures of the SIP/IP Core, as specified in [3GPP TS 24.229];
10. SHALL check if each of the invited user(s) are not already listed among the active Participants of the CPM Group Session, or if an outstanding invitation was already sent to the invited user(s) and is pending a response. If any of these are true, then the CPM Controlling Function SHALL not continue with the next step;
11. SHALL follow the actions described in section 9.2.10 "*Create Session with a Participant*" for each address.

9.2.6 Removing Participant Request

CPM SIP REFER requests are sent within the scope of the CPM Session dialog to which they are related. Upon receiving a SIP REFER request that includes an Accept-Contact header with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' and the "method" parameter set to "BYE" in the Refer-To header, the CPM Controlling Function:

1. SHALL if the Refer-Sub header is not present or is set to "true" in the SIP REFER request, generate and send a SIP 421 "Extension Required" response including a Require header with the option tag "norefersub".

   Note: any network entities along the path which act as a B2BUA SHALL populate the Refer-Sub header field with a value of "false".
2. SHALL perform actions to verify that the authenticated originator's CPM Address of the CPM User who initiated the request is authorized for the request;
   a. in case of CPM Pre-defined Group, by checking the <allow-expelling> element as defined in [OMA-XDM-Group];
   b. in all other cases based on service provider policy;
   c. if not authorized, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "122 Function not allowed" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of steps;
3. SHALL check whether privacy is allowed for the authenticated originator's CPM Address, when anonymity is requested with the Privacy header containing the tag 'id'. If not allowed, the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "119 Anonymity not allowed" in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of steps;
4. SHALL extract from the SIP REFER request the address(es) of the user(s) to be removed either:
   a. from the Refer-To header according to the rules and procedures of [RFC3515] and [RFC7647], when removing one user; or
   b. from the MIME resource-list body indicated by the Content-ID header that has same value as the CID in the Refer-To header field, according to the rules and procedures of [RFC5368] for removing multiple users.

5. SHALL generate a SIP 200 “OK” response to the SIP REFER request according to the rules and procedures of [RFC7647]
6. SHALL include in the response to the SIP REFER request a Refer-Sub header set to “false” according to the rules and procedures of [RFC4488];
7. SHALL include in the response to the SIP REFER request a Supported header with the option tag “norefersub” according to the rules and procedures of [RFC4488];
8. SHALL follow the actions described in section 9.2.11 “*Participant Removing Request*” for each address;

9.2.7 Session Leaving Request

Upon receiving a SIP BYE request with a Reason header field containing a reason code=200 with text “Call completed”, the CPM Controlling Function SHALL consider the Participant “departed” and it:

1. SHALL return a SIP 200 "OK" response towards the Participant according to the rules and procedures of the SIP/IP core;
2. SHALL release all Media Plane resources corresponding to the CPM Session being closed for the Participant being removed and discard any locally queued media that was not yet sent to the Participant;
3. SHALL terminate the subscription to Participant Information of this CPM Group Session as specified in section 9.2.14.4 “*Terminating the Subscription*” if the CPM Client requesting to leave the CPM Group Session has such a subscription;
4. SHALL send SIP NOTIFY to the remaining participants of the chat indicating by the value “departed” in the <disconnection-method> associated with that CPM User;
5. SHALL remove the Participant from the CPM Group Session Participants Information;
6. SHALL check CPM Group Session release policy as specified below:
    a. if a CPM Group Session for a CPM Pre-defined Group or a CPM Ad-hoc Group has as many as, or less than specified Participants left, the CPM Controlling Function SHALL remove any remaining Participant from the CPM Session according to Section 9.2.11 “*Participant Removing Request* ” and release the CPM Session. The possible values should be 0 or 1.
    b. If the CPM Session is a CPM Ad-hoc Group that is not Long-lived and the originator of the CPM Session leaves the CPM Session then the CPM Controlling Function SHALL remove the rest of the Participants from the CPM Session it hosts according to Section 9.2.11 “*Participant Removing Request*” and release the CPM Session;
7. If Long-lived CPM Group Session is supported, subject to service provider policies, the CPM Controlling Function SHALL keep locally the information of the Group State Object (including the type of CPM Group Session i.e. Closed, and the list of participants) to allow the Participants to re-start it at a later time.

If there is on-going CPM File Transfer initiated in the CPM Group Session, the termination of the CPM Group Session SHALL not terminate the on-going CPM File Transfer for the CPM Participants in the CPM Group Session, and vice versa.

If the CPM Controlling Function received a SIP BYE with Reason header field including a SIP reason with a cause value different than protocol-cause=200, then the CPM Controlling Function SHALL consider the Participant “booted” and it:

1. SHALL send a 200 OK response towards the Participant according to the rules and procedures of the SIP/IP core;
2. SHALL release the Media Plane resources associated with the Participant and keep the Participant in the CPM Group Session Participants Information;
3. SHALL record the ‘booted’ value in the <disconnection-method> for the Participant in the Participants Information of the CPM Group Session;
4. SHALL send SIP NOTIFY to the remaining participants of the chat indicating by the value “booted” in the <disconnection-method> associated with that CPM User, with a reason sub-element set to SIP response code 5*03 <reason>SIP;cause=503;text="Service Unavailable"</reason>.*

9.2.8 CPM Group Session Modification

Upon receiving a SIP re-INVITE request (i.e. an INVITE request with an existing CPM Session Identity in the Request-URI, including a new SDP offer) the CPM Controlling Function:

1. SHALL validate that the new media parameters in the SIP re-INVITE request are acceptable by the CPM Controlling Function according to service provider policy. If not acceptable SHALL return a SIP 488 "Not Acceptable Here" response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
2. In case the CPM Group Session has been established on behalf of a CPM Pre-defined Group, SHALL perform actions to verify that the authenticated originator’s CPM Address of the CPM User who initiated the request is authorized for the request by checking the <allow-media-handling> element as defined in [OMA-XDM-Group] according to CPM Pre-defined Group policy. If not authorized, the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include a SIP Warning header with the warning text set to “122 Function not allowed” in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of steps;
3. SHALL send a session modification request to the other Participants of the CPM Group Session. The CPM Controlling Function:
   a. SHALL generate a SIP re-INVITE request according to the rules and procedures of [RFC3261];
   b. SHALL include a MIME SDP body based on the received SDP offer, as described in section 5.2.1.2 “*SDP Handling at Intermediate Nodes*”;
   c. SHALL send the SIP re-INVITE request towards the CPM Group Session Participants according to the rules and procedures of SIP/IP core.
4. SHALL determine based on service provider policy the minimum number of Participants that must accept the session modification in order to allow the modification to proceed:
   a. If enough Participants have agreed to the session modification, SHALL generate a SIP 200 "OK" response as follows:
      i. Include a MIME SDP body as a SDP answer as described in section 5.2.1.3 “*SDP Handling at Terminating Nodes*” with the following addition:
         1. If the CPM Controlling Function supports the CPM Group Session Data management , then it SHALL populate in the a=accept-wrapped-types attribute the value of the CPM Group Session Data Content-Type as defined in section 6.8 "*CPM Group Session Data Management*" and the value “multipart/related”;
         2. If the CPM Controlling Function supports Participant Information Notifications via Conference Information Object, then it SHALL include in the a=accept-wrapped-types attribute the value “application/conference-info+xml”, and the value “multipart/related” (if not already present);
      ii. Send the SIP 200 "OK” response to the SIP/IP core along the signalling path.
   b. If not enough Participants have accepted the session modification, SHALL send a 488 “Not Acceptable Here” response and send a SIP re-INVITE request to those Participants that had accepted the session modification, in order to bring the session back to the original session media parameters.

      NOTE: Service provider policy determines how to handle the sessions when different participants accept different media types.

Upon receiving a SIP ACK request, the CPM Controlling Function SHALL modify the Media Plane for all CPM Group Session Participants who accepted the CPM Session modification as specified in section 9.2.12 “*Media Plane Handling*”.

9.2.9 Group Session Ending Request

When the CPM Controlling Function finds that the CPM Group Session SHOULD be released based on the CPM release policy described below, the CPM Controlling Function SHALL remove all the Participants in the CPM Group Session as specified in section 9.2.11 “*Participant Removing Request*” , with the following clarifications:

- If the number of participants is smaller than the minimum number required for a CPM Group Session, then the Controlling Function SHALL indicate this by including a Reason header field with a value set to *SIP;cause=410;* and MAY have a text parameter containing an explanatory String (e.g. ”Gone”, or other) in the SIP BYE request that it sends to the remaining participants, to indicate that the CPM Group Session Identity no longer exists;
- If the CPM Group Session inactivity timer was reached, the Controlling Function SHALL include a Reason header field with a value set to *SIP;cause=480;* and MAY have a text parameter containing an explanatory String

(e.g.*"Bearer unavailable", or other)* in the SIP BYE request that it sends to the current participants, indicating to the CPM Clients that the CPM Group Session can re-started as a CPM Long-lived group Session.

The CPM Controlling Function SHALL support the following operator configurable variables as defined in [OMA-XDM-Policy] together with the possible values shown in parentheses:

auto-release (true/false)

1. When the initiator leaves the CPM Session and auto-release is true the CPM Controlling Function SHALL remove the rest of the Participants from the CPM Group Session for a CPM Pre-defined Group and release the CPM Session;
2. If false the CPM Controlling Function SHALL NOT remove the rest of the Participants from the CPM Group Session for a CPM Pre-defined Group nor release the CPM Group Session when the initiator leaves the CPM Session.

session-max-length (seconds)

1. If the CPM Group Session has lasted the specified amount of seconds the CPM Controlling Function SHALL release the CPM Group Session.

number-of-remaining-participants (0/1)

1. If a CPM Group Session for a CPM Pre-defined Group or a CPM Ad-hoc Group has as many as or less than the specified Participants left the CPM Controlling Function SHALL release the CPM Group Session. The possible values are 0 or 1.

   NOTE: Service provider policy defines the values of these variables for CPM Ad-Hoc Group Sessions.

The release policy SHALL include the following:

1. The CPM Controlling Function SHALL remove the rest of the Participants from the CPM Group Session it hosts and release the CPM Group Session:
   a. If the CPM Group Session is an Ad-hoc Group Session and the initiator of the CPM Group Session leaves and the service provider policy requires closure of the session in such circumstances;
   b. If the CPM Group Session is a CPM Pre-defined Group Session and the initiator of the CPM Group Session leaves and the auto-release has the value "true";
   c. If the CPM Group Session is a CPM Pre-defined or Ad-hoc Group Session and there are as many as or less than the "number-of-remaining-participants" Participants in the CPM Group Session; or
   d. If the CPM Group Session has lasted longer than the session-max-length variable specifies;

9.2.10 Create Session with a Participant

When the CPM Controlling Function needs to set up a session with a CPM Group Session Participant, CPM Controlling Function:

1. SHALL generate an initial SIP INVITE request according to the rules and procedures of [RFC3261] with the following clarifications:
2. SHALL include an Accept-Contact header field with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' if an Accept-Contact header field has been received in the incoming CPM Session Invitation to start the CPM Group Session but does not contain a 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' CPM Feature Tag, according to the rules and procedures of [RFC3841];
3. SHALL copy the values received in each Accept-Contact header field of the incoming CPM Session Invitation to start the CPM Group Session to a corresponding Accept-Contact header field in the SIP INVITE request, if any Accept-Contact header fields are received;
4. SHALL include in the Contact header of the SIP INVITE request as CPM Group Session Identity the contact address of the conference and the CPM feature-tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session' and the "isfocus" feature parameter;
5. SHALL include the Session-Expires header field with the refresher parameter set to 'uas' according to the rules and procedures of [RFC4028];
6. SHALL include the Supported header field set to "timer";
7. SHALL include value 'id' in a Privacy header field according to the rules and procedures of [RFC3325], if anonymity is requested with the "Privacy: id" header field in the incoming CPM Session Invitation to start the CPM Group Session;

8. SHALL include a MIME SDP body as a SDP offer based on the received SDP offer as described in section 5.2.1.2 "*SDP Handling at Intermediate Nodes*", with the following addition:
    a. If the CPM Controlling Function supports the CPM Group Session Data management , then it SHALL populate in the a=accept-wrapped-types attribute the value of the CPM Group Session Data Content-Type as defined in section 6.8 "*CPM Group Session Data Management*" and the value "multipart/related", in the SDP offer;
    b. If the CPM Controlling Function supports Participant Information Notifications via Conference Information Object, then it SHALL include in the a=accept-wrapped-types attribute the value "application/conference-info+xml" and the value "multipart/related" (if not already present);
9. SHALL include the authenticated originator's CPM Address as specified in section 6.1 "*Authenticated Originator's CPM Address*" as follows:
    a. with the URI and Display name set to the CPM Group Session Identity in the case of a CPM Group Session for a CPM Ad-hoc Group Session; or,
    b. with the URI and Display name set to the CPM Pre-defined Group address in the case of a CPM Group Session for a CPM Pre-defined Group;
10. SHALL set the Request-URI to the CPM Address of the Participant to be invited;

    NOTE: If the request is extending an existing CPM 1-1 Session to a CPM Group Session, the Request-URI may contain the public GRUU of the CPM Client on which the CPM Session is being extended.
11. SHALL include a Referred-By header field with the authenticated originator's CPM Address present in the incoming CPM Session Invitation to start the CPM Group Session or in the REFER request when a participant is added to an ongoing session, as specified in section 6.1 "*Authenticated Originator's CPM Address*";
12. SHALL include the Conversation-ID, Contribution-ID and InReplyTo-Contribution-ID header fields and values received in the incoming CPM Session Invitation to start the CPM Group Session;
13. MAY include the "Subject" header field associated with the group if it is a CPM Pre-defined Group or in the incoming CPM Session Invitation to start a CPM Ad-Hoc Group or a subject was stored in the Group State Object for that group if the session is created with an existing participant of the CPM Group Session. If the session is created with a new participant to the group (e.g. following an Adding Participants Request as specified in section 9.2.5), a "Subject" header field SHALL be added if a subject is available from the CPM Pre-defined Group definition, the incoming CPM Session Invitation to start a CPM Ad-Hoc Group or the stored Group State Object. If no subject is available, the "Subject" header field SHALL NOT be included;

    NOTE: The Subject header field of the CPM Pre-defined Group has precedence over the Subject header field received in the incoming CPM Session Invitation to start the CPM Group Session.
14. SHALL include a resource-list-history body in the outgoing SIP INVITE request, if the SIP INVITE request is sent in reaction to an incoming SIP INVITE request, containing the addresses of the invited participants in the received SIP INVITE request (e.g. as a consequence of a SIP REFER, it is populated with the active and pending participants) and SHALL set the Content-Disposition header field to "recipient-list-history; handling=optional" as described in [RFC5366];

    NOTE: "To", "CC" and "BCC" qualifiers are not considered since those are of little relevance in the context of CPM Sessions.
15. SHALL send the SIP INVITE request towards the Participant, via SIP/IP core according to the rules and procedures of the SIP/IP core and consider the Participant in a "pending" state.
16. Subject to service provider policies, the CPM Controlling Function MAY start buffering MSRP messages exchanged by the other Participants in the CPM Group Session until a 2xx final response is received from the pending Participant. In such case, the buffered CPM Chat messages are delivered to the CPM Participant when the MSRP session gets established. If a SIP error response is received instead of a 2xx final response, the CPM Controlling Function SHALL discard any buffered messages for the Participant.

Upon receiving an error response that indicates that the recipient is not a CPM subscriber, e.g. a SIP 488 "Not Acceptable Here" response, the CPM Controlling Function SHALL

1. generate and send a SIP ACK request according to the rules and procedures of [RFC3261]
2. send the SIP INVITE request to the ISF as described in section 6.5 "Communicating with the ISF and IWF" if the response is on of
    a. 404 "Not Found" or

   b. 405 “Method Not Allowed” or
   c. 410 “Gone” or
   d. 414 “Request URI Too Long” or
   e. 415 “Unsupported Media Type” or
   f. 416 “Unsupported URI Scheme” or
   g. 488 “Not Acceptable Here” or
   h. 606 “Not Acceptable Here”.

   Responses coming from the IWF are treated the same way as responses coming from CPM Participants, but will never be considered to be a reason to send the request to the ISF.

3. MAY send a notification to the CPM Clients, which have subscribed to the Participant information that an pending CPM User has left the CPM Group Session, as specified in section 9.2.14.3 *“Sending Participant Information Notifications"*.

Upon receiving SIP 200 "OK" response for the SIP INVITE request the CPM Controlling Function SHALL

1. check whether privacy is allowed in this CPM Group, when anonymity is requested with the Privacy header containing the tag "id". If not allowed, the CPM Controlling Function SHOULD release the CPM Client from the CPM Session immediately as specified in 9.2.11 “Participant Removing Request”. Allowing privacy in the group is defined using <allow-anonymity> element as specified in [OMA-XDM-Group]. If allowed, provide anonymization according to [RFC3323] and [RFC3325] and continue with the rest of the steps;
2. when a SIP ACK request was already received from the initiating CPM user,
   a. generate and send a SIP ACK request to the CPM Group member from which the SIP Final response was received; and
   b. initiate the Media Plane as defined in section 9.2.12 “Media Plane Handling”
   c. forward any buffered MSRP messages to the CPM Group member when the Media Plane is established
3. consider the added participant to be in the “Connected” state and SHALL NOTIFY the other participants of the updated conference state.

If the CPM Controlling Function received a SIP 603 DECLINED response to a SIP INVITE, then it SHALL remove the Participant that has declined from the Participants list and SHALL notify the remaining participants of an updated conference state. In this conference state event package notification, the CPM Controlling Function SHALL set the departed participant’s status to “disconnected” and include the disconnection-method element set to a value of “failed” with a reason sub-element set to code 603 <reason>SIP;cause=603;text="Declined"</reason>.

9.2.11 Participant Removing Request

When a Participant needs to be removed from the CPM Group Session, the CPM Controlling Function SHALL remove the Participant as follows:

If the CPM Controlling Function received a request to remove the Participant, via a SIP REFER request procedure described in sect. 9.2.6 “*Removing Participant Request*“ or decides to terminate the session by removing the remaining participants as described in section 9.2.9 “*Group Session Ending Request*”, then it:

1. SHALL generate a SIP BYE request according to the rules and procedures of [RFC3261]. In this SIP BYE request:
   a. SHALL include a with a Reason header field containing SIP;cause=410; and MAY have a text parameter containing an explanatory String (e.g. "Gone", or other),
   b. MAY include Referred-by header field set to the authenticated originator’s CPM Address of the participant that requested the removing request;
   c. SHALL send the SIP BYE request towards the Participant according to the rules and procedures of the SIP/IP core as defined in [3GPP TS24.229];
2. SHALL release the Media Plane resources associated with the Participant being removed discarding also any media that remains to be sent to the user and keep the Participant in the CPM Group Session Participants Information;
3. Upon receiving a SIP 200 "OK" or any other final response for the SIP BYE request, the CPM Controlling Function
   a. SHALL send a notification to the Participants that have subscribed to the Group Session information, to indicate that a CPM User has left the CPM Group Session, as specified in section 9.2.14.3 “*Sending*

Participant Information Notifications" indicating the value "departed" in the <disconnection-method> associated with that CPM User, and

b. SHALL terminate the subscription to Participant Information of this CPM Group Session as specified in section 9.2.14.4 "*Terminating the Subscription*" if the CPM Client requesting to leave the CPM Group Session has such a subscription.

9.2.12 Media Plane Handling

The CPM Controlling Function bridges Media Streams for CPM Group Session Participants.

9.2.12.1 Media Plane Handling for MSRP Sessions

The CPM Controlling Function relays MSRP Media Streams between originating network and terminating networks. In general, the CPM Controlling Function:

1. SHALL maintain a MSRP session for each participant who accepted the MSRP session;
2. SHALL distribute received MSRP messages to each participant who accepted the MSRP session, except for the MSRP SEND messages that include the CPM Group Session Data Content-Type.

To establish the MSRP connection, the CPM Controlling Function:

1. SHALL act as MSRP client for sending MSRP SEND request according to [RFC6135];
2. SHALL act as an "active" or "passive" endpoint as negotiated and according to[RFC6135] based on the received SDP offer and answer; and
3. SHALL establish the MSRP connection towards each participant who accepted the MSRP session according to the MSRP connection parameters in the SDP answer received in the 200 "OK" response according to[RFC6135];

Upon establishment of the MSRP connection for a session with a Participant for which Participant Information Notifications via Conference Information Object is applicable, the CPM Controlling Function SHALL notify the CPM Client about the Participant List, <subject>, <bulletin> and <icon> data via a MSRP SEND request as defined in section 10.2.14.3 *"Sending Participant Information Notifications"*.

Upon receiving an MSRP SEND request, the CPM Controlling Function:

1. SHALL generate and send MSRP 200 OK response to the MSRP SEND request;
2. SHALL, if the received Content-type in MSRP is CPIM and the CPIM message content or the CPIM header defined in Appendix C.1.12 "*Payload-Type*" does not include the content type of the CPM Group Session Data management defined in section 6.8 *"CPM Group Session Data Management"*, duplicate the MSRP SEND request for each CPM Group member that can receive the message according to the SDP negotiation done at session set up (e.g. max-size and wrapped content-type) with the following modifications according to the rules and procedures of [RFC4975]:

   NOTE: the duplication and subsequent sending may start as soon as all MSRP headers and for the first chunk of the message, the CPIM headers have been received.

   a. SHALL modify the To-Path header according to the MSRP URI(s) received in the answer SDP from the CPM Group member in accordance with rules and procedures of [RFC4975];

   b. SHALL modify the From-Path header to the CPM Controlling Function's own MSRP URI, according to the rules and procedures of [RFC4975];

   c. SHALL send the MSRP SEND requests towards each participant who accepted the MSRP session (both CPM Users and/or non-CPM Principals) excluding the one that sent the MSRP SEND request via established MSRP connections taking into account the maximum chunk size negotiated according to section 5.2.1, if any;

   d. MAY temporarily queue the MSRP SEND request for participants that have not yet provided a final response to the SIP INVITE request to set up the session.
3. SHALL, if the CPIM message includes the CPM Group Session Data Management content type defined in section 6.8 *"CPM Group Session Data Management"*, check if the Group Session Data management request is valid by checking that the CPIM body conforms with the CPM Group Session Data Management structure and schema and

by checking the data received in the CPIM headers, according to the section 6.8 *"CPM Group Session Data Management"* and if the Participant is authorized to make that CPM Group Session Data operation for that CPM Group Session Identity, then:

a. If the request is valid and authorized, the CPM Controlling Function:

    i. SHALL update the CPM Group Session information with the received changes for <subject> or for <icon> data or the <role> data of a Participant;

    ii. SHALL generate notifications containing the updated information to the Participants, as described in section 9.2.14.3 "*Sending Participant Information Notifications*";

b. otherwise, if the request is not valid, or is not authorized, the CPM Controlling Function SHALL generate an error response to the CPM Client;

c. SHALL send a response back to the CPM Client in a MSRP SEND request containing the result of the operation in the <response> element, according to CPM Group Session Data <response> element structure as defined in section 6.8.1 "*Data Structure of the CPM Group Session Data Management*", and populating the appropriate CPIM headers as defined in section 6.8 *"CPM Group Session Data Management"*;

d. SHALL NOT forward the received MSRP SEND to the other Participants.

4. SHALL temporarily buffer any messages sent when one or more of the participants are in the "pending" state (i.e. no SIP final response has been received on the SIP INVITE request to add them to the session).

   NOTE: those buffered messages will be forwarded to the participant when they accept the session as detailed in section 9.2.10.

9.2.12.2 Media Plane Handling for RTP/RTSP Sessions

The CPM Controlling Function SHALL establish RTP sessions and relay RTP and RTCP packets between originating network and terminating networks according to the rules and procedures of [RFC3550]. In general, the CPM Controlling Function:

1. SHALL maintain a RTP session for each CPM Group member who accepted the RTP session;
2. SHALL relay or distribute received RTP and RTCP to each CPM Group member who accepted the RTP session.

9.2.13 Pseudonyms in a CPM Group Session

The CPM Controlling Function maintains information about Pseudonyms of all the Participants in a session. The purpose is to guarantee that a Pseudonym is unique in a session.

When the CPM Controlling Function receives a SIP INVITE request and the CPM User requests anonymity by adding privacy type "id" to the Privacy header as defined in [RFC3325] or the CPM Controlling Function receives a SIP REFER request that is without a method parameter in the Refer-To header or when the method parameter is set to "INVITE" in the Refer-To header and the CPM User requests anonymity by adding privacy type "id" to the Privacy header as defined in [RFC3325], the CPM Controlling Function SHALL check the 'From' header field values against the entries in the CPM Controlling Function mapping for the session and

1. If the 'From' header field values are unique, the CPM Controlling Function SHALL store the 'From' header field values for the joining User.
2. If the 'From' header field is NOT unique, the CPM Controlling Function SHALL assign a modified/changed values of the 'From' header field to make it unique for the joining User.

   NOTE: When an CPM Controlling Function checks the 'From' header field e.g. <bob><sip: anonymous@anonymous.invalid>" and this field is not unique in the CPM Group Session, then the CPM Controlling Function could modify it to <bob-X><sip:anonymousX@anonymous.invalid>, where X could be any value. This is an example and the exact form of modified Pseudonym is a matter of service provider policy.

If the CPM user is subscribed to the Participant Information notification, the CPM Controlling Function:

1. SHALL insert in the Participant Information destined to the joining User, the modified value of the 'From' header field as the value of the "entity" attribute of the <user> element;
2. SHALL include "yourown" attribute extension with the value "true" to the <user> element.

9.2.14 Participant Information

9.2.14.1 CPM Group Session Participant Information Request

Upon receiving a SIP SUBSCRIBE request with the Event header set to “conference” according to the rules and procedures of [RFC4575], the CPM Controlling Function:

1. SHALL check whether the SUBSCRIBE contains a SIP Event header set to “conference”. If not, CPM Controlling Function SHALL return a SIP 489 “Bad Event” response and SHALL include a SIP Warning header with the warning text set to “122 Function not allowed”

   Otherwise continue with the rest of the steps;
2. SHALL, if the Request-URI contains a CPM Group Session Identity, check whether the CPM Group Session Identity in the Request-URI is owned by the CPM Controlling Function. If not, the CPM Controlling Function SHALL return a SIP 404 “Not found” response;

   Otherwise continue with the rest of the steps;
3. SHALL verify the authenticated originator’s CPM Address and authorize the request according to CPM Pre-defined Group policy and/or service provider policy and if it is not authorized the CPM Controlling Function SHALL return a SIP 403 "Forbidden" response and SHALL include an SIP Warning header with the warning text set to “127 Service not authorised” in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;

   NOTE: Service provider policy, e.g. number of simultaneous subscriptions exceeded, may cause the CPM Controlling Function to reject the subscription request.
4. In case of CPM Ad-hoc Group, SHALL check if the requesting CPM User is a Participant. If not, the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include an SIP Warning header with the warning text set to “122 Function not allowed” in the response according to the rules and procedures of [RFC3261]. A CPM Participating Function serving a Participant SHALL be accepted by the CPM Controlling Function as a valid subscription on behalf of a Participant (e.g. CPM Participating Function subscribes to the conference event package in the deferred CPM Group Session scenario).

   Otherwise, continue with the rest of the steps;
5. In case of CPM Pre-defined Group, SHALL check whether the authenticated originator’s CPM Address is allowed to subscribe to Participant information according to the <allow-conference-state> action as specified in [OMA-XDM-Group] and if user is a Participant in the CPM Group Session. If it is not allowed or if the requesting CPM User is not a Participant, the CPM Controlling Function SHALL return a SIP 403 “Forbidden” response and SHALL include an SIP Warning header with the warning text set to “122 Function not allowed” in the response according to the rules and procedures of [RFC3261].

   Otherwise, continue with the rest of the steps;
6. MAY check whether there is an active subscription from the requesting user and if so, MAY terminate that subscription as specified in section 9.2.14.4 “*Terminating the Subscription*”.
7. SHALL create a subscription to the conference state of the CPM Group Session according to the rules and procedures of [RFC6665] and [RFC4575] and SHALL generate a SIP 200 "OK" response to the SIP SUBSCRIBE request according to the rules and procedures of [RFC6665] and [RFC4575];
8. SHALL set the Contact header of the SIP response to the address of the CPM Group Session Identity;
9. SHALL send the SIP response according to the rules and procedures of the SIP/IP core;
10. SHALL generate and send an initial SIP NOTIFY request as specified in section 9.2.14.3 “*Sending Participant Information Notifications*”;
11. If the CPM Group Session is still active, the CPM Controlling Function keeps the subscription and further provides conference state updates.
    a) When a change in the subscribed state occurs, the CPM Controlling Function SHOULD generate and send a SIP NOTIFY request as specified in section 9.2.14.3 “*Sending Participant Information Notifications*” and according to the rules and procedures of SIP/IP core respectively.
12. When needed the CPM Controlling Function SHALL terminate the subscription and indicate it to the CPM Client as described in the section 9.2.14.4 “*Terminating the Subscription*”.

9.2.14.2 Long-lived CPM Group Session

The Long-lived CPM Group Session requires the CPM Controlling Function to keep the conference information related to the CPM Group Session after the CPM Group Session became inactive, for a duration that is subject to service provider policies. This allows the CPM Controlling Function to further serve requests related to the Long-lived CPM Group Session (e.g. re-start of Long-lived CPM Group Session).

The minimum conference information kept by the CPM Controlling Function is listed below. The CPM Controlling Function:

1) SHALL keep:
    a. Conversation-ID, Contribution-ID and CPM Group Session Identity;
    b. last Participants Information;
    c. type of the CPM Group Session (e.g. Closed or not);
2) if it supports the CPM Group Session Data management, then it SHALL keep:
    a. the subject, with the Participant that has last updated it and the timestamp of the update, and
    b. the icon, with the Participant that has last updated it and timestamp of the update,
    c. the bulletin, with the Participant that has last updated it and timestamp of the update, and
    d. roles assigned to Participants.

9.2.14.3 Sending Participant Information Notifications

When the CPM Controlling Function needs to notify a Participant of changes to the Participant Information, and a subscription to the conference state of the CPM Group Session exists for a participant, then it SHALL generate a SIP NOTIFY request according to the rules and procedures of [RFC6665]with the following clarifications:

1. When reporting changes in the Participant information, the CPM Controlling Function SHALL use partial notification according to the rules and procedures of [RFC4575];
2. The CPM Controlling Function SHALL include a MIME application/conference-info+xml body according to the rules and procedures of [RFC4575]. The conference-information body:
    a. SHALL include a <conference-info> element. The <conference-info> element:
    b. SHALL include the CPM Group Session Identity in the <entity> attribute of the <conference-info> element;
    c. MAY include the <state> attribute with the values "full" or "partial" according to the rules and procedures of [RFC4575];
    d. SHALL include the <version> attribute according to the rules and procedures of [RFC4575];
    e. SHALL include the <conference-description> element. The <conference-description> element:
        i. if CPM Group Session Data Management is supported and if a CPM ad hoc Group Policy applies, SHALL include the <keywords> elements with the values as defined in Appendix Q. *"CPM Ad-hoc Group Policy"*.
        ii. if CPM Group Session Data Management is supported and if a bulletin has been set for the CPM Group Session, SHALL include the <free-text> elements including the bulletin text.
        iii. SHALL include the <maximum-user-count> element indicating the maximum number of participants allowed in the CPM Group Session.
    f. SHALL include the <conference-state> element. The <conference-state> element
        i. SHALL include the <user-count> element indicating the number of Participants in the CPM Group Session at the moment.
    g. SHALL include the <users> element. The <users> element
        i. MAY include the <state> attribute with the values "full" or "partial" according to the rules and procedures of [RFC4575].
    h. SHALL include a <user> element per Participant in the CPM Group Session. The "user" element:
    i. SHALL include a <users> element with a <user> sub element for each participant in the session. The "user" element:
        i. SHALL include the <entity> attribute. The <entity> attribute:
            1. SHALL for the originating CPM Client include the authenticated originator’s CPM Address of the initial SIP INVITE request, if the Participant has not requested privacy; or,

2. SHALL for the originating CPM Client include the from header, if the Participant has requested privacy; and,
3. SHALL for an invited CPM Client include the identity used in the URI-list for the Invited CPM Client to a CPM Ad-hoc Group Session or the identity used in the CPM Group definition in case of a Predefined Group Session, if the Participant has not requested privacy; or,
4. SHALL for an invited CPM Client include an anonymous identity as specified in [RFC4575], if the Participant has requested privacy;
5. SHALL include, for each Participant receiving the NOTIFY an extension to the <user> element, “yourown” attribute to identify the Participant’s own information as defined in Appendix M of [OMA-SIMPLE-IM].

ii. MAY include the <display-text> element. If included, the <display-text> element SHALL include the Display Name of the identity which was used in the <entity> attribute as defined in step a;

iii. if CPM Group Session Data Management is supported and if a Participant is assigned with a CPM User role, SHALL include a <roles>element with one or more <entry> elements, each indicating a role of a CPM User as defined in Appendix Q.1.2 *"CPM User Roles"*.

iv. SHALL include a single <endpoint> element. The <endpoint> element:

1. SHALL include the <entity> attribute with the same value as the <entity> attribute in the <user> element;
2. SHALL include the <status> element. The <status> element has one of the following values:
   - a. SHALL support "connected", when the Participant has joined to the CPM Group Session; or,
   - b. SHALL support "disconnected", when the Participant has left the CPM Group Session since the last SIP NOTIFY request was sent;
   - c. MAY support “pending”, when the acceptance of the CPM Group Session invitation was not yet received from the Participant as per RFC4575);
3. if the <status> element has the value "disconnected", SHALL include the <disconnection-method> element with one of the values "departed”, “booted” or “failed”;

v. If it supports the CPM Group Session Data management, then it SHALL include:

1. the <subject> element. If no subject is assigned to the CPM Group Session, then an empty value SHALL be used;
2. if the timestamp of the update of the <subject> element is newer than the time the Participant receiving the SIP NOTIFY was added to the CPM Group Session or if the Participant receiving the SIP NOTIFY initiated the CPM Group Session initially, the <subject-ext> element as defined in Appendix 0 *"CPM Extensions to the Conference Event Package"* with the <timestamp> element indicating the time when the change was made and then <participant> element indicating the participant that made the change;
3. the <icon> element as defined in Appendix 0 *"CPM Extensions to the Conference Event Package"*. The <icon> element:
   - a. if no icon is assigned to the CPM Group Session, SHALL NOT include a <source> element, otherwise
   - b. SHALL include a <source> element. The <source> element SHALL include one of the following child elements:
     - i. <icon-uri> element of type anyURI containing the URI pointing to the image. The procedure for the CPM Client to determine a URI for the icon image is out of scope of this specification. If no icon is assigned to the CPM Group Session, then the <icon-uri> element SHALL contain an empty value.
     - ii. <file-info> element of type String containing a content-id (CID) pointing to an additional body of the SIP NOTIFY containing the image file.

c. if the timestamp of the update of the <icon> element is newer than the time the Participant receiving the SIP NOTIFY was added to the CPM Group Session or if the Participant receiving the SIP NOTIFY initiated the CPM Group Session initially, SHALL include

i. the <timestamp> element indicating the time when the change was made, and

ii. the <participant> element indicating the participant that made the change;

4. The <bulletin> element.

Next, The CPM Controlling Function SHALL send the SIP NOTIFY request according to the rules and procedures of the SIP/IP core.

The CPM Controlling Function SHOULD limit the rate of SIP NOTIFY requests sent towards a Participant.

NOTE: How a CPM Controlling Function limits the rate of SIP NOTIFY requests sent towards the CPM Client is out of scope of this specification.

When the CPM Controlling Function needs to notify a Participant of changes to the Participant Information and;

A. a subscription to the conference state of the CPM Group Session does not exist; and,

B. Participant Information Notifications via Conference Information Object is applicable for the CPM Group Session with a participant,

it SHALL generate an MSRP SEND request according to the procedures described in section 9.2.12.1 "*Media Plane Handling for MSRP Sessions*" with the difference that the Content-Type type of the CPIM wrapped message body SHALL be set to "application/conference-info+xml" body and that the schema defined in [RFC4575] with the CPM extensions defined in Appendix P "*The CPM Group Session Data Management*" are used for the XML data structure included in the CPIM body, as specified below. When reporting changes in the Participant Information, the CPM Controlling Function SHALL NOT use partial notification, each Conference Information object sent to the CPM Client SHALL contain the coherent state of the CPM Group Session. The conference-info body:

a. SHALL include a <conference-info> element. The <conference-info> element

i. SHALL include the CPM Group Session Identity in the <entity> attribute;

ii. SHALL NOT include the <state> attribute.

iii. SHALL NOT include the <version> attribute.

iv. SHALL include the <conference-description> element. The <conference-description> element:

1) if CPM Group Session Data Management is supported, SHALL include the <subject> element indicating the subject of the CPM Session. If no subject is assigned to the CPM Group Session, then the <subject> element SHALL contain an empty value.

2) if the <subject> element is present, and if the timestamp of the update of the <subject> element is newer than the time the Participant receiving the MSRP SEND was added to the CPM Group Session or if the Participant receiving the MSRP SEND initiated the CPM Group Session initially, SHALL include <subject-ext> element as defined in Appendix P.2 *"CPM Extensions to the Conference Event Package"* with the <timestamp> element indicating the time when the change was made and then <participant> element indicating the participant that made the change.

3) if CPM Group Session Data Management is supported and if a CPM ad hoc Group Policy applies, SHALL include the <keywords> elements with the values as defined in Appendix Q. *"CPM Ad-hoc Group Policy"*.

4) if CPM Group Session Data Management is supported and if a bulletin has been set for the CPM Group Session, SHALL include the <free-text> elements including the bulletin text.

5) SHALL include the <maximum-user-count> element indicating the maximum number of participants allowed in the CPM Group Session.

6) if CPM Group Session Data Management is supported the <icon> element as defined in Appendix P.2 *"CPM Extensions to the Conference Event Package"*. The <icon> element:

a. if no icon is assigned to the CPM Group Session, SHALL NOT include a <source> element, otherwise

b. SHALL include a <source> element. The <source> element SHALL include one of the following child elements:

i. <icon-uri> element of type anyURI containing the URI pointing to the image. The procedure for the CPM Client to determine a URI for the icon image is out of scope of this specification. If no icon is assigned to the CPM Group Session, then the <icon-uri> element SHALL contain an empty value.

ii. <file-info> element of type String containing a content-id (CID) pointing to an additional body of the MSRP SEND containing the image file in this case, the Content-Type in the MSRP body SHALL be a MIME “multipart/related” Content-Type, including both the Content-Type “application/conference-info+xml” and the Content-Type of the file, and the cpm.Payload-Type header shall be included, listing both “application/conference-info+xml” and the Content-Type of the file.

c. if the timestamp of the update of the <icon> element is newer than the time the Participant receiving the MSRP SEND was added to the CPM Group Session or if the Participant receiving the MSRP SEND initiated the CPM Group Session initially, SHALL include:

i. the <timestamp> element indicating the time when the change was made, and

ii. the <participant> element indicating the participant that made the change;

v. SHALL include the <conference-state> element. The <conference-state> element:

1) SHALL include the <user-count> element indicating the number of Participants in the CPM Group Session at the moment;

vi. SHALL include the <users> element. The <users> element:

1) SHALL NOT include the <state> attribute.

2) SHALL include a <user> element per Participant in the CPM Group Session. The "user" element:

a. SHALL include the <entity> attribute. The <entity> attribute:

i. SHALL for the originating CPM Client include the authenticated originator’s CPM Address of the initial SIP INVITE request, if the Participant has not requested privacy; or,

ii. SHALL for the originating CPM Client include the from header, if the Participant has requested privacy; and,

iii. SHALL for an invited CPM Client include the authenticated originator’s CPM Address of the response to the initial SIP INVITE request or include the identity used in the URI-list for the Invited CPM Client to a CPM Ad-hoc Group Session or the identity used in the CPM Group definition in case of a Predefined Group Session, if the Participant has not requested privacy; or,

iv. SHALL for an invited CPM Client include an anonymous identity as specified in [RFC4575], if the Participant has requested privacy;

v. SHALL include, for each Participant receiving the MSRP SEND request an extension to the <user> element, “yourown” attribute to identify the Participant’s own information as defined in Appendix M of [OMA-SIMPLE-IM].

b. MAY include the <display-text> element. If include, the <display-text> element SHALL include the Display Name of the identity which was used in the <entity> attribute as defined in step a;

c. if CPM Group Session Data Management is supported and if a Participant is assigned with a CPM User role, SHALL include a <roles>element with one or more <entry> elements, each indicating a role of a CPM User as defined in Appendix Q.1.2 "*CPM User Roles*".

d. SHALL include a single <endpoint> element. The <endpoint> element:

i. SHALL include the <entity> attribute with the same value as the <entity> attribute in the <user> element;

ii. SHALL include the <status> element. The <status> element has one of the following values:

a. SHALL support "connected", when the Participant has joined to the CPM Group Session; or,

b. SHALL support "disconnected", when the Participant has left the CPM Group Session since the last MSRP SEND request carrying a Content-Type or cpm.Payload-Type with value application/conference-info+xml sent while this CPM Group Session has been active.

c. MAY support “pending”, when the acceptance of the CPM Group Session invitation was not yet received from the Participant as per RFC4575).

d. if the <status> element has the value "disconnected", SHALL include the <disconnection-method> element with one of the values departed”, “booted” or “failed”;

e. if the <status> element has the value "disconnected", SHALL include the <disconnection-info> element. The <disconnection-info>:

   1. SHALL include the <when> element .

   2. MAY include the <reason> element. The <reason> element:

      a. if the Participant was removed via a SIP REFER request procedure described in section 9.2.6 “*Removing Participant Request*“, SHALL include "SIP;cause=410"; and MAY have a text parameter containing an explanatory String (e.g. "Gone", or other);

      b. if the Participant departed the CPM Group Session via a SIP BYE request procedure described in section 9.2.7 "Session Leaving Request", SHALL include "SIP;cause=200;text="Call completed".

   3. MAY include the <by> element. If the Participant was removed via a SIP REFER request procedure described in section 9.2.6 “Removing Participant Request“, then the <by> element SHALL include the authenticated originator’s CPM Address of the participant that requested the removing request.

f. if the <status> element has the value "connected", SHALL include the <joining-info> element if the participant was added via a SIP REFER request procedure described in section 9.2.5 "Adding Participants Request". The <joining-info>:

   1. SHALL include the <when> element including the date and time that the Participant was added to the CPM Session;

   2. MAY include the <reason> element. The <reason> element:

      a. if the Participant was removed via a SIP REFER request procedure described in section 9.2.6 “*Removing Participant Request*“, SHALL include "SIP;cause=410"; and MAY have a text parameter containing an explanatory String (e.g. "Gone", or other);

      b. if the Participant departed the CPM Group Session via a SIP BYE request procedure described in section 9.2.7 "Session Leaving Request", SHALL include "SIP;cause=200;text="Call completed"".

         i. MAY include the <by> element. If the participant was added via a SIP REFER request procedure described in section 9.2.5 “Adding Participants Request“, then the <by> element SHALL include the authenticated originator’s CPM Address of the participant that requested the adding request.

9.2.14.4 Terminating the Subscription

The CPM Controlling Function:

1. SHALL terminate all subscriptions for Participant information for the CPM Group Session when the CPM Group Session is released and only accept new subscriptions for Long-lived CPM Group Sessions;
2. SHALL terminate the subscription for an CPM Client when it leaves the CPM Session;

3. For each subscription that is terminated the CPM Controlling Function as described in sections 9.2.7 "*Session Leaving Request*" and 9.2.11 "*Participant Removing Request*":
    a. SHALL generate a SIP NOTIFY request as described in section 9.2.14.3 "*Sending Participant Information Notifications*" according to the rules and procedures specified in [RFC6665];
    b. SHALL set the Subscription-State header to "terminated; reason=noresource"; and,
    c. SHALL send the SIP NOTIFY request to the CPM Client according to the rules and procedures of SIP/IP core.

9.2.15 Disposition Notification

When the CPM Controlling Function receives a CPM Standalone Message, or a CPM Chat Message, containing a disposition notification request, the CPM Controlling Function SHALL act as intermediary according to the rules and procedures of [RFC5438] with the following clarifications:

- In case the CPM Controlling Function needs to stay on the path of IMDN to aggregate IMDNs according to service provider policy, it SHALL insert in the CPM Standalone Message, or in the CPM Chat Message, a CPIM IMDN-Record-Route header with its own address;
- The CPM Controlling Function SHALL insert in the CPM Standalone Message, or in the CPM Chat Message, a CPIM Original-To header with the value of the original CPIM To header;
- In case the CPM Controlling Function receives an error response to a CPM Standalone Message, or to the CPM Chat Message, containing a "negative-delivery" notification request, the CPM Controlling Function MAY construct and send a "negative-delivery" notification according to service provider policy.

When staying on the path of IMDN to aggregate IMDNs, upon receiving the initial IMDN, the CPM Controlling Function SHALL await for any remaining IMDNs, aggregate them and send the aggregated IMDN according to rules and procedures of [RFC5438].

9.3 CPM File Transfer Handling

When the CPM Controlling Function receives a SIP INVITE request with the CPM Feature Tag 'urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer', it SHALL execute the processing in section 9.1.2 "*Large Message Mode CPM Standalone Message*".

NOTE: The SDP for Large Message Mode CPM Standalone Message and CPM File Transfer is different but the processing in the CPM Controlling Function is the same.

9.3.1 CPM File Transfer in Group Session Handling

If File Transfer in CPM Group Session is supported , the CPM Controlling Function receives a SIP INVITE request with CPM File Transfer Feature Tag and with the Request-URI having a value of the CPM Group Session Identity of the on-going active CPM Group Session, it SHALL execute the procedures in section 9.1.2 "*Large Message Mode CPM Standalone Message*" with the following clarifications:

1. SHALL handle the CPM Participants that did not indicate support for CPM supported MIME types in the *a=accept-wrapped-types* attribute of the SDP answer to the CPM File Transfer SIP INVITE, as per service provider policy; and

2. SHALL not include the recipient-list in the body of the generated INVITE requests (i.e. skip the steps 3 b, c and d, and step 4 in section 9.1.2 "*Large Message Mode CPM Standalone Message*").

If File Transfer in CPM Group Session is not supported and the CPM Controlling Function receives a SIP INVITE request with CPM File Transfer Feature Tag and with the Request-URI having a value of the CPM Group Session Identity of the on-going active CPM Group Session, it SHALL reject the SIP INVITE request with a SIP 403 "Forbidden" response and SHALL include a SIP Warning header with the warning text set to "122 Function not allowed" in the response according to the rules and procedures of [RFC3261].

The resumption of CPM File Transfer is not supported for the Group CPM File Transfer case. The CPM Controlling Function SHALL NOT relay a CPM File Transfer resume request to the CPM User that was the original initiator of the corresponding CPM File Transfer. The CPM Controlling Function SHALL send a SIP 488 NOT ACCEPTABLE HERE error response.

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
n/a	n/a	No prior version

A.2 Draft/Candidate Version 2.2 History

Document Identifier	Date	Sections	Description
Draft Versions OMA-TS-CPM_Conversation_Function-V2_2	01 Mar 2017	All	Initial Draft with the content from OMA-TS-CPM_Conversation_Function-V2_1-20170222-D document
	09 May 2017	All	Incorporate AGREED CRs: • OMA-COM-CPM-2017-0033R01-CR_Participant_Information_Notifications_in_Media_Plane • OMA-COM-CPM-2017-0048R01-CR_Missing_client_direct_delivery_procedure_for_standalone_messaging • OMA-COM-CPM-2017-0053-CR_C021_ContributionID_in_IMDN • OMA-COM-CPM-2017-0054R01-CR_CONRR_C003_Payload_Type_CPIM_header • OMA-COM-CPM-2017-0055-CR_CONRR_C004_ToHeader_inGroupChat • OMA-COM-CPM-2017-0056R02-CR_CONRR_C007_C023_C025_PF_orig_term • OMA-COM-CPM-2017-0057-CR_CONRR_C016_GSMA_alignments • OMA-COM-CPM-2017-0078R01-CR_Version_2_2_summary • OMA-COM-CPM-2017-0059-CR_C002_CPIM_notif • OMA-COM-CPM-2017-0060R01-CR_C022_cid_REFER • OMA-COM-CPM-2017-0061-CR_C006_C008_client_procedures_group_session_mgmt • OMA-COM-CPM-2017-0062-CR_C010_C024_TermPF • OMA-COM-CPM-2017-0063-CR_C011_PF_IMDN • OMA-COM-CPM-2017-0070-CR_C026_multipart_in_NOTIFY • OMA-COM-CPM-2017-0076R01-CR_C015_number_participants • OMA-COM-CPM-2017-0077R01-CR_Support_Aliasing_Function_UP2.0
	10 May 2017	Editorials	Editorial fixes: – Date on cover page updated (March-> May) – Remove duplicate text in sections 5.4.1, 5.4.6 – Remove leftover comment in section 6.7.3.6. – formatting error, fixed missing last bullet in section 7.2.3.1 – fixed numbering issue at the end of section 7.3.1.1 (bullet list started at 4 instead of 1) – removed empty bullet in section 9.2.1 – fixed numbering issues in section 9.2.14.3
	26 May 2017	Several sections	Incorporated CRs: • OMA-COM-CPM-2017-0081-CR_Bug_fixes_CPIM_wrapper
	27 May 2017	6.8, 7.3.9.1, 9.2.12, Appendix Q.1.	Incorporated CR: OMA-COM-CPM-2017-0021R06-CR_User_Role_Management

Document Identifier	Date	Sections	Description
	04 Jun 2017	2.1, 6.8, 9.2.12.1, Appendix C.1.14.	Incorporated CRs: • OMA-COM-CPM-2017-0083-CR_Correction_of_CPIM_header_handling_for_Group_Data_Management • OMA-COM-CPM-2017-0084-CR_Correction_of_CPIM_Payload_Type_header
	12 Jun 2017	9.2.12.1, 9.2.14.3	Incorporates agreed CR: • OMA-COM-CPM-2017-0085R02-CR_CPIM_header_alignment_with_CR83
	14 Sep 2017	Several sections	Incorporated CRs: • OMA-COM-CPM-2017-0092-CR_ConvTS_clarifications_alignment_RCS_UP_EndorsementRCC.11 • OMA-COM-CPM-2017-0093-CR_Bug_Fix_in_Section_9.1.1
Candidate Version OMA-TS-CPM_Conversation_Function-V2_2	26 Sep 2017	n/a	Status changed to Candidate by TP TP Ref # OMA-TP-2017-0040-INP_CPM-V2_2_ERP_for_1st_Candidate_Approval

Appendix B. Static Conformance Requirements (Normative)

The notation used in this appendix is specified in [OMA-SCRRULES].

B.1 SCR for CPM Client

Item	Function	Reference	Requirement
CPM-CF-C-001-M	Registering	7.1	
CPM-CF-C-002-M	Sending a Pager Mode CPM Standalone Message to one recipient	7.2.1.1	
CPM-CF-C-003-M	Sending a Pager Mode CPM Standalone Message to a CPM Ad-hoc Group	7.2.1.1	
CPM-CF-C-004-M	Sending a Pager Mode CPM Standalone Message to a CPM Pre-defined Group	7.2.1.1	
CPM-CF-C-005-M	Sending a Large Message Mode CPM Standalone Message to one recipient	7.2.1.2	
CPM-CF-C-006-M	Sending a Large Message Mode CPM Standalone Message to a CPM Ad-hoc Group	7.2.1.2	
CPM-CF-C-007-M	Sending a Large Message Mode CPM Standalone Message to a CPM Pre-defined Group	7.2.1.2	
CPM-CF-C-008-O	Forwarding/Including Stored Data without Downloading to the CPM Client – Pager Mode	7.2.1.4	
CPM-CF-C-009-O	Forwarding/Including Stored Data without Downloading to the CPM Client – Large Message Mode	7.2.1.4	
CPM-CF-C-010-M	Receiving Pager Mode CPM Standalone Message	7.2.2.1	
CPM-CF-C-011-M	Receiving Large Message Mode CPM Standalone Message	7.2.2.2	
CPM-CF-C-012-M	Subscribing to Deferred CPM Message Information	7.2.3.1	
CPM-CF-C-013-M	Handling selected Deferred CPM Messages before expiry	7.2.3.2.1	
CPM-CF-C-014-M	Retrieving all Deferred CPM Messages before expiry	7.2.3.2.1	
CPM-CF-C-015-O	Retrieving CPM Standalone Messages after expiry	7.2.3.2.2	
CPM-CF-C-016-O	Handling expiry notifications for Deferred CPM Messages	7.2.3.3	
CPM-CF-C-017-M	Out-of-band notification handling	7.2.3.4.1	
CPM-CF-C-018-M	In-band notification handling	7.2.3.4.2	

Item	Function	Reference	Requirement
CPM-CF-C-019-M	Generate Delivery Notification	5.4.1	
CPM-CF-C-020-M	Generate Read Report	2	
CPM-CF-C-021-M	Receive Delivery Notification	5.4.3	
CPM-CF-C-022-M	Receive Read Report	5.4.4	
CPM-CF-C-023-M	Initiating a CPM 1-1 Session	7.3.1.1	
CPM-CF-C-021-M	Initiating a CPM Group Session for a CPM Ad-hoc Group	7.3.1.2	
CPM-CF-C-022-M	Initiating a CPM Group Session for a CPM Pre-defined Group	7.3.1.3	
CPM-CF-C-023-M	Join a CPM Group Session for a Join-in Group	7.3.1.4	
CPM-CF-C-024-M	Receiving a CPM Session Initiation	7.3.2	
CPM-CF-C-025-M	Extending a CPM 1-1 Session to a CPM Group Session	0	
CPM-CF-C-026-M	Sending a CPM 1-1 Session Closure	7.3.4.1	
CPM-CF-C-027-M	Leaving a CPM Group Session	7.3.4.2	
CPM-CF-C-028-M	Receiving a CPM Session Closure	7.3.4.3	
CPM-CF-C-029-M	Inviting one User to existing CPM Group Session	7.3.5	
CPM-CF-C-030-M	Inviting more than one User to existing CPM Group Session	7.3.5	
CPM-CF-C-031-M	Removing one Participant from a CPM Group Session	7.3.6	
CPM-CF-C-032-O	Removing more than one Participant from a CPM Group Session	7.3.6	
CPM-CF-C-033-M	Modifying a CPM Session	7.3.7	
CPM-CF-C-034-M	CPM Client receiving an CPM Session modification request	7.3.8	
CPM-CF-C-035-M	CPM Session Media Plane Handling – MSRP	7.3.9.1	
CPM-CF-C-036-M	CPM Session Media Plane Handling – RTP/RTCP	7.3.9.2	
CPM-CF-C-037-M	Subscribe to Receiving Participant Information	7.3.10.1	
CPM-CF-C-038-M	Receive Participant Information Notification	7.3.10.2	
CPM-CF-C-039-M	Initiating a CPM File Transfer to a single recipient	7.4.1	
CPM-CF-C-040-M	Initiating a CPM File Transfer to a CPM Ad-hoc Group	7.4.1	
CPM-CF-C-041-M	Initiating a CPM File Transfer to a CPM Pre-defined Group	7.4.1	

Item	Function	Reference	Requirement
CPM-CF-C-042-M	Receiving a CPM File Transfer	7.4.2	
CPM-CF-C-043-M	Releasing a CPM File Transfer	7.4.3	
CPM-CF-C-044-M	Fetching Deferred CPM File Transfer file(s)	7.4.4	
CPM-CF-C-045-M	Resuming an interrupted CPM File Transfer	7.4.5	
CPM-CF-C-046-M	Handling “isComposing”	5.5	
CPM-CF-C-047-M	Handling Deferred CPM Group Session message delivery	7.3.12	
CPM-CF-C-048-M	Handling Deferred CPM 1-1 Session message delivery	7.3.11	
CPM-CF-C-049-M	Handling Re-joining CPM Long-live Group Session	7.3.1.5	
CPM-CF-C-050-M	Handle CPM Session Closed Group Chat	7.3.1	CPM-CF-C-021-M AND CPM-CF-C-030-M

B.2 SCR for CPM Participating Function

Item	Function	Reference	Requirement
CPM-CF-PF-S-001-M	Handle registration events	8.1	
CPM-CF-PF-S-002-M	Handle Pager Mode CPM Standalone Message in originating network	8.2.1.1	
CPM-CF-PF-S-003-O	Fetch and include objects from Message Storage Server	8.2.1.1	
CPM-CF-PF-S-004-O	Extend Pager Mode to Large Message Mode	8.2.1.1	
CPM-CF-PF-S-005-M	Handle Large Message Mode CPM Standalone Message in the originating network	8.2.1.2	
CPM-CF-PF-S-006-O	Fetch and include objects from Message Storage Server	8.2.1.2	
CPM-CF-PF-S-007-M	Handle Disposition Notification in the originating network	8.2.4, 8.2.4.1	
CPM-CF-PF-S-008-M	Handle a CPM Session Invitation in the originating network	8.2.2.1	
CPM-CF-PF-S-009-M	Handle a SIP CANCEL request in the originating network	8.2.2.2	
CPM-CF-PF-S-010-M	Handle a SIP BYE request in the originating network	8.2.2.3	
CPM-CF-PF-S-011-M	Handle SIP session expiry in the originating network	8.2.2.4	
CPM-CF-PF-S-012-M	Handle CPM Session modification request in the originating network	8.2.2.5	

Item	Function	Reference	Requirement
CPM-CF-PF-S-013-M	Handle CPM File Transfer initiation in the originating network	8.2.3.1	
CPM-CF-PF-S-014-M	Handle CPM File Transfer closing in the originating network	8.2.3.2	
CPM-CF-PF-S-015-M	Handle Pager Mode CPM Standalone Message in the terminating network	8.3.1.1	
CPM-CF-PF-S-016-M	Handle Large Message Mode CPM Standalone Message in the terminating network	8.3.1.2	
CPM-CF-PF-S-017-O	Replacing Media with a reference	8.3.1.4	
CPM-CF-PF-S-018-M	Sending disposition notifications	8.3.1.6	
CPM-CF-PF-S-019-M	Defer CPM Standalone Messages	8.3.1.6	CPM-CF-PF-S-020-O OR CPM-CF-PF-S-021-O OR CPM-CF-PF-S-022-O
CPM-CF-PF-S-020-O	Pushing Deferred CPM Messages	8.3.1.6.2	
CPM-CF-PF-S-021-O	Sending notifications and awaiting CPM Client action	8.3.1.6.3	
CPM-CF-PF-S-022-O	Delivering Deferred CPM Message to the Message Storage Server	8.3.1.6.4	
CPM-CF-PF-S-023-M	Handle Deferred Message Information Request	8.3.1.6.5	
CPM-CF-PF-S-024-M	Handling Deferred CPM Messages on expiry time	8.3.1.6.8	
CPM-CF-PF-S-025-O	Sending expiry notifications	8.3.1.6.8	
CPM-CF-PF-S-026-M	Handle a CPM Session Invitation in the terminating network	8.3.2.1	
CPM-CF-PF-S-027-M	Handle a SIP CANCEL request in the terminating network	8.3.2.1.2	
CPM-CF-PF-S-028-M	Handle a SIP BYE request in the terminating network	8.3.2.3	
CPM-CF-PF-S-029-M	Handle SIP session expiry in the terminating network	8.3.2.4	
CPM-CF-PF-S-030-M	Handle CPM Session modification request in the terminating network	8.3.2.5	
CPM-CF-PF-S-031-M	Handle CPM File Transfer initiation in the terminating network	8.3.3.1	
CPM-CF-PF-S-032-M	Handle CPM File Transfer closing in the terminating network	8.3.3.2	
CPM-CF-PF-S-033-O	Record CPM Standalone	8.5.1	

Item	Function	Reference	Requirement
	Message		
CPM-CF-PF-S-034-O	Record CPM Session	8.5.2	
CPM-CF-PF-S-035-O	Record CPM File Transfer	8.5.3	
CPM-CF-PF-S-036-M	Generate Delivery Notification	5.4.1	
CPM-CF-PF-S-037-M	Handling CPM Long-lived Group Chat	8.2.2.7.2	CPM-CF-PF-S-010-M AND CPM-CF-PF-S-038-O AND CPM-CF-PF-S-039-M
CPM-CF-PF-S-038-O	Handle Participant Information	8.2.2.6	
CPM-CF-PF-S-039-M	Handle CPM Group Session Re-join Request	8.2.2.7	
CPM-CF-PF-S-040-M	Handle Deferred delivery CPM Group Session	0	CPM-CF-PF-S-038-O
CPM-CF-PF-S-041-M	Handle Deferred CPM File Transfer Files(s)	8.3.3.3	CPM-CF-PF-S-042-O AND CPM-CF-PF-S-043-O AND CPM-CF-PF-S-044-O AND CPM-CF-PF-S-045-O AND CPM-CF-PF-S-046-O
CPM-CF-PF-S-042-O	Delivering Deferred CPM File Transfer file(s)	8.3.3.4	
CPM-CF-PF-S-043-O	Generate request for Deferred delivery of CPM File Transfer File(s)	8.3.3.4	
CPM-CF-PF-S-044-O	Receive request for Deferred delivery of CPM File Transfer File(s)	8.3.3.4	
CPM-CF-PF-S-045-O	Storing CPM File Transfer Deferred Information	8.3.3.7	
CPM-CF-PF-S-046-O	Media Plane handling for MSRP Session	8.6	

B.3 SCR for CPM Controlling Function

Item	Function	Reference	Requirement
CPM-CF-CF-S-001-M	Pager Mode CPM Message to CPM Ad-Hoc Group	9.1.1	
CPM-CF-CF-S-002-M	Pager Mode CPM Message to CPM Pre-defined Group	9.1.1	
CPM-CF-CF-S-003-M	Large Message Mode CPM Message to CPM Ad-Hoc Group	9.1.2	
CPM-CF-CF-S-004-M	Large Message Mode CPM Message to CPM Pre-defined Group	9.1.2	
CPM-CF-CF-S-005-M	Disposition Notification Handling	9.2.15	
CPM-CF-CF-S-006-M	Ad-hoc Session Invitation Request	9.2.1	
CPM-CF-CF-S-007-M	Pre-defined Group Session Invitation Request	9.2.1	

Item	Function	Reference	Requirement
CPM-CF-CF-S-008-M	Session Cancellation Request	9.2.2	
CPM-CF-CF-S-009-M	Joining Pre-defined Group Session Request	9.2.3	
CPM-CF-CF-S-010-M	Rejoining Group Session Request	9.2.4	
CPM-CF-CF-S-011-M	Adding Participants Request	9.2.5	
CPM-CF-CF-S-012-M	Removing Participant Request	9.2.6	
CPM-CF-CF-S-013-M	Session Leaving Request	9.2.7	
CPM-CF-CF-S-014-M	Group Session Modification	9.2.8	
CPM-CF-CF-S-015-M	Group Session Ending Request	9.2.9	
CPM-CF-CF-S-016-M	Group Session Media Plane Handling – MSRP	9.2.12.1	
CPM-CF-CF-S-017-M	Group Session Media Plane Handling – RTP/RTCP	9.2.12.2	
CPM-CF-CF-S-018-O	Pseudonyms in a CPM Group Session	9.2.13	
CPM-CF-CF-S-019-M	CPM Group Session Information Request	9.2.14.1	
CPM-CF-CF-S-020-M	Sending Participant Information notifications	9.2.14.3	
CPM-CF-CF-S-021-M	Terminating the subscription	9.2.14.4	
CPM-CF-CF-S-022-M	CPM File Transfer Handling	9.3	
CPM-CF-CF-S-023-M	Handling CPM Long-lived Group Chat	9.2.4	CPM-CF-CF-S-010-M AND CPM-CF-CF-S-015-M AND CPM-CF-CF-S-024-M
CPM-CF-CF-S-024-M	Participant Removing Request	9.2.11	
CPM-CF-CF-S-025-M	Handling CPM Session Closed Group Chat	8.2.1	CPM-CF-CF-S-006-M AND CPM-CF-CF-S-007-M AND CPM-CF-CF-S-008-M AND CPM-CF-CF-S-009-M AND CPM-CF-CF-S-010-M AND CPM-CF-CF-S-011-M AND CPM-CF-CF-S-013-M

Appendix C. CPM-defined Header fields (Normative)

C.1 Header field Definitions

C.1.1 Conversation-ID

A Conversation-ID header field in a SIP MESSAGE request or SIP INVITE request indicates the CPM Conversation Identity associated with a CPM Standalone Message, CPM File Transfer, or CPM Session.

A sending CPM functional component MUST include a Conversation-ID header field in each SIP MESSAGE request or SIP INVITE request that are associated with a CPM Standalone Message, a CPM File Transfer, or a CPM Session. The sending CPM functional component MUST ensure that the included CPM Conversation Identity is globally unique. A Conversation-ID SHALL be a universally unique identifier, as specified in [RFC4122]. Following the conventions in [RFC4122], Conversation-IDs are case-insensitive.

Examples:

```
Conversation-ID: f81d4fae-7dec-11d0-a765-00a0c91e6bf6
```

C.1.2 Contribution-ID

A Contribution-ID header field in a SIP MESSAGE request or SIP INVITE request indicates the CPM Contribution Identity associated with a CPM Standalone Message, CPM File Transfer, or CPM Session.

A sending CPM functional component MUST include a Contribution-ID header field in each SIP MESSAGE request or SIP INVITE request that are associated with a CPM Standalone Message, a CPM File Transfer, or a CPM Session. The sending CPM functional component MUST ensure that the included CPM Contribution Identity is unique within the context of the encompassing CPM Conversation. A Contribution-ID SHALL be a universally unique identifier, as specified in [RFC4122]. Following the conventions in [RFC4122], Contribution-IDs are case-in sensitive.

Example:

```
Contribution-ID: dfb4e64f-2c98-46a1-8e14-98eb5736a268
```

C.1.3 InReplyTo-Contribution-ID

An InReplyTo-Contribution-ID header field in a SIP MESSAGE request or SIP INVITE request indicates the CPM Contribution Identity associated with a CPM Standalone Message, CPM File Transfer, or CPM Session, to which this request is a reply to.

When a CPM User replies to a CPM Standalone Message, CPM File Transfer, or CPM Session, the replying CPM Client MUST include a InReplyTo-Contribution-ID header field in the SIP MESSAGE request or SIP INVITE request associated with the reply CPM Standalone Message, the reply CPM File Transfer or the reply CPM Session, and populate it with the CPM Contribution Identity of the CPM Standalone Message, CPM File Transfer or CPM Session being replied to.

Example:

```
InReplyTo-Contribution-ID: 01234567-89ab-cdef-0123-456789abcdef
```

C.1.4 Session-Replaces

A Session-Replaces header field in a SIP INVITE request carries the same CPM Contribution Identity associated with a previously established CPM 1-1 Session that should be replaced by the new CPM Session Invitation.

A sending CPM Client MUST include a Session-Replaces header field in the SIP INVITE request that extends a CPM 1-1 Session into a CPM Group Session. This header field will contain the value of the Contribution-ID sent in the original CPM 1-1 Session. The receiving CPM Client uses this value to end the previous session if it accepts the new session invitation.

Example:

```
Session-Replaces: abcdef-1234-5678-90ab-cdef01234567
```

C.1.5 Message-Expires

A Message-Expires header field in a SIP INVITE request to initiate a Large Message Mode CPM Standalone Message transfer carries the expiry time associated with the Large Message Mode CPM Standalone Message. It functions in the same manner as the Expires header field for MESSAGE requests.

NOTE: A Message-Expires header field in a SIP INVITE request that is not meant to initiate a Large Message Mode CPM Standalone Message transfer is to be ignored by the SIP UAS.

A sending CPM Client MUST include a Message-Expires header field in each SIP INVITE request that starts the transfer of a Large Message Mode CPM Standalone Message if it wants to associate an expiry time with that Large Message Mode CPM Standalone Message. This header field will contain the relative time after which the Large Message Mode CPM Standalone Message expires.

Example:

```
        Message-Expires: 259200
```

C.1.6 Message-UID

A Message-UID header field in a SIP MESSAGE request, 200 OK SIP response or SIP BYE request indicates the Unique Identifier value as described in [OMA-CPM_TS_MessageStorage] of the associated CPM Standalone Message or CPM File Transfer History that is stored in the CPM User’s network-based Message Storage.

When a CPM Standalone Message or a CPM File Transfer History is stored by the CPM Participating Function in the CPM User’s network-based Message Storage, the delivering CPM Participating Function MUST include a Message-UID header field in each SIP MESSAGE request, 200 OK response or SIP BYE request associated with a CPM Standalone Message, or a CPM File Transfer History.

Examples:

```
        Message-UID: 4392
```

When a CPM Standalone Message includes this header field, the CPM Client SHALL store the value of this Message-UID header field in conjunction with the received CPM Standalone Message or CPM File Transfer History in a local storage. When synchronizing the messages on the locally stored in a device with the CPM User’s Message Storage Server, the CPM Standalone Message object or CPM File Transfer History object having this UID value will not be retrieved again.

C.1.7 Contact Header field values

The Contact header field in a SIP INVITE request and responses may carry the ‘isfocus’ feature parameter as described in [RFC4579].

Any SIP entity in the path SHALL copy the Contact header field transparently if the ‘isfocus’ feature parameter is contained.

C.1.8 SDP attributes and values

The SDP a-line attribute “a=max-chunk-size” includes the value of the proposed MSRP chunk size on either SDP offerer or SDP answerer sides, as per its own configured values based on service provider policies (e.g. for the network access type, or the applicable service level agreement). The value in the SDP answer is the maximum MSRP chunk size to be used by the MSRP endpoints in the SIP session for the duration of the session.
The recommended minimum value in Kbytes SHOULD be 100 KB.

C.1.9 Message-Direction

The CPIM Message-Direction header field indicates whether the message has been originally sent or received by the CPM User. In a multi-device environment, this header field contains an indication for other CPM User’s Clients (except the one that sent the CPM Chat Message) to display the sending chat message correctly in a UI chat window.

The header is defined as an extension to the [RFC5322] field definitions. The limits for the occurrence of the field are defined in the following table:

Field	Min Number	Max Number
Message-Direction	0	1

Table 4 - Message-Direction header

The field itself is defined in ABNF as follows:

```
message-direction    = "Message-Direction:" message-direction-part CRLF
message-direction-part = message-direction-value / extension
message-direction-value           = "sent" / "received"
extension             = *CHAR
```

Example:

```
Message-Direction: sent
```

A client receiving the Message-Direction header with a value other than "sent" or "received" shall ignore the header.

C.1.10 Message-Context

A Message-Context SIP header field is used in a 200 “OK” response sent by the CPM IWF to any one of the following:

- a SIP MESSAGE request to initiate interworking of a Pager Mode CPM Standalone Message transfer, or
- A SIP INVITE request to initiate to initiate interworking of a CPM Session; or
- a SIP BYE request, closing the SIP session where the CPM IWF has successfully delivered an interworked CPM Large Message Mode Standalone Message, or an interworked CPM File Transfer,

The Message-Context SIP header field carries the information on the non-CPM technology that was used to deliver that CPM Message, or CPM File Transfer, to a CPM Client that resides on a Primary Device without SIP/IP core connectivity. It is used by CPM IWF to provide the interworking technology to the CPM Participating Function, for storage purposes.

It has the syntax of the Message-Context MIME header defined in [RFC3458] and can take one of the following values:

1. pager-message: when the successful delivery was done via SMS; or
2. multimedia-message: when successful delivery was done via MMS

Example:

```
Message-Context: pager-message
Message-Context: multimedia-message
```

C.1.11 Message-Correlator

A Message-Correlator SIP header field is used in a 200 “OK” response sent by the CPM IWF to provide the MMS message id value used for the interworked delivery of the message to MMS:

- a SIP BYE request, closing the SIP session where the CPM IWF has successfully delivered an interworked CPM Large Message Mode Standalone Message, or an interworked CPM File Transfer,

It carries correlation information that SHOULD support a CPM Client to match a message received via the non-CPM technology with the equivalent message object retrieved from the CPM Message Store.

The Message-Correlator SIP header field is used by CPM IWF to provide the MMS Message ID of the message delivered via interworking to MMS to the CPM Participating Function, for storage purposes. The value of the received Message-Correlator SIP header field is set by the CPM Participating Function in the MIME header Message-Correlator of the object delivered via interworking to MMS.

The syntax and ABNF of the Message-Correlator is as defined in Appendix C.6 of the [OMA-CPM_TS_MessageStorage].

C.1.12 IW-Number

An IW-Number SIP header field is used in a SIP request (MESSAGE or INVITE) and in their SIP 200 “OK” response and it SHALL contain the value of the unique interworking number allocated to interworking to SMS or MMS of a Long-lived CPM Group Session.

Originally added by ISF in SIP requests sent to the selected CPM IWF that is performing either SMS or MMS interworking of the Long-lived CPM Group Session, the IW-Number SIP header field is also provided by ISF and IWF in a SIP 200 “OK” response back to the CPM Participating Function. The CPM Participating Function uses the value of the IW-Number SIP header field to populate the “iw-number” attribute in the Group State Object(s) stored in the Long-lived CPM Group Session history recorded in the CPM Message Store for the CPM User to whom the CPM Chat Messages of that Long-lived CPM Group Session have been delivered via interworking.

The IW-Number SIP header field is of type “anyURI” and SHALL contain the unique interworking number as a TEL URI.

Example:

```
IW-Number: tel:+15145550001
```

C.1.13 CPM namespace for CPIM headers extensions

[RFC3862] defines a framework for CPIM header extensibility whose use is optional. As per [RFC3862] the extensibility framework is based on that provided for XML by XML namespaces.

This specification defines a CPIM namespace for the CPIM extension headers defined by CPM. The namespace is:

“urn:oma:cpm:cpim”

As per CPIM [RFC3862] requirements, the new header fields defined by the CPM specification within the CPM CPIM namespace are prepended, in CPIM messages, by a prefix assigned to the URN through the NS header field of the CPIM message. This specification always assumes an NS header field like this one:

NS: cpm <http://www.openmobilealliance.org/cpm/cpim >

Note: As per [RFC3862] the clients are free to use any prefix while servers and intermediaries must accept any legal namespace prefix specification.

This CPM namespace association applies to all CPM defined CPIM headers. Following basic XML rules to ensure header names uniqueness, when used in CPIM messages the CPM defined CPIM headers are always prepended with their associated CPM namespace. Within a particular message instance, the header names are associated with a particular namespace through the presence or absence of a namespace prefix, which is a leading part of the header name followed by a period ("."), for example:

cpm.header-name: header-value

where ‘cpm’ is the header name prefix designating the namespace, 'header-name' is the header name within the namespace associated with 'cpm', and 'header-value' is the value for this header.

Note: per XML rules, when the ‘cpm’ namespace is absent then the given 'header-name' is associated with the default namespace (in the CPIM case i.e. <urn:ietf:params:cpim-headers:> as defined in [RFC3862]).

C.1.14 Payload-Type

The CPIM Payload-Type header field contains an enumeration of the Content-Types included in the payload of a CPM Message.

The payload of user messages is typically conveyed in the CPIM message content. If the user message consists of multiple parts, then the CPIM message includes a multipart entity. Any CPM entity that needs to know the Content-Types of the individual body parts of the multipart entity would need to open up and parse the CPIM message payload in order to detect the included Content-Types. To avoid this, the Content-Types of the body parts of a multipart entity SHALL be listed in the CPIM Payload-Type header field, to be quickly accessible without requiring to open up the payload.

The header field is defined as an extension to the [RFC3862] field definitions. The limits for the occurrence of the field are defined in the following table:

Field	Min Number	Max Number
Payload-Type	0	1

Table 5 - Message-Direction header

The field itself is defined in [RFC 3862] ABNF as follows:

```
      Payload-type = "Payload-Type:" SP payload-type-part CRLF
      payload-type-part = content-type-value *(";" SP content-type-value)
      content-type-value = type "/" subtype
                          ; for the definition of type and subtype refer
                          ; to [RFC2045]
```

Example:

```
      cpm.Payload-Type: text/plain; audio/mpeg; image/jpeg
```

Where `content-type` is defined in [RFC3458].

C.2 ABNF for the CPM-defined SIP Headers

```
CPM-Headers = (Conversation-ID
            /  Contribution-ID
            /  InReplyTo-Contribution-ID
            /  Session-Replaces
            /  Message-Expires
            /  Message-UID)  CRLF

Conversation-ID = "Conversation-ID" HCOLON conversationid
conversationid = word

Contribution-ID = "Contribution-ID" HCOLON contributionid
contributionid = word

InReplyTo-Contribution-ID = "InReplyTo-Contribution-ID" HCOLON
                              contributionid

Session-Replaces = "Session-Replaces" HCOLON sessionreplaces
Sessionreplaces = word

Message-Expires = "Message-Expires" HCOLON delta-seconds

Message-UID = "Message-UID" HCOLON uniqueid

Message-Context ABNF notation is defined in the [RFC3458].

Message-Correlator ABNF notation is defined in the Appendix C.6 of the [OMA-
CPM_TS_MessageStorage].

IW-Number = telephone-uri, where telephone-uri syntax and ABNF notation are defined
in [RFC3966].
```

NOTE 1: The tokens “CRLF”, “HCOLON”, “word” and “delta-seconds” are defined in [RFC3261].

NOTE 2: The token “uniqueid” is defined in [RFC3501].

C.3 ABNF for the CPM extensions to SIP Headers

This specification allows having several URI values in the Referred-By header field(s), extending the definition provided in [RFC3892].

A Referred-By header field value SHALL consist of exactly one ‘name-addr’ or ‘addr-spec’. There MAY be one or two Referred-By values. If there is one value, it SHALL be a sip, sips, or tel URI. If there are two values, one value SHALL be a sip or sips URI and the other SHALL be a tel URI.

Referred-By header field BNF from [RFC3892]:

```
Referred-By  =  ("Referred-By" / "b") HCOLON referrer-uri

*( SEMI (referredby-id-param / generic-param) )
```

This specification extends the [RFC3892] BNF as follows:

```
Referred-By  =  ("Referred-By" / "b") HCOLON referrer-uri
( SEMI (referredby-id-param / generic-param) )


referrer-uri = ( name-addr / addr-spec )

referredby-id-param = add-refs / cid

add-refs = "add-refs" EQUAL LDQUOT ( name-addr / addr-spec ) RDQUOT

cid = "cid" EQUAL sip-clean-msg-id

sip-clean-msg-id = LDQUOT dot-atom "@" (dot-atom / host) RDQUOT

        dot-atom = atom *( "." atom )

            atom     = 1*( alphanum / "-" / "!" / "%" / "*" /"_" / "+" / "'" / "`"
      / "~")
```

Note: cid parameter is not used in this specification.

Example:

```
               Referred-By: <sip:charlie@tele.com>; add-refs="tel:+123456789"
```

C.4 ABNF for the CPM-defined SDP parameter

The ABNF syntax for the SDP a-line attribute “a=max-chunk-size” is:

```
max-chunk-size = max-chunk-size-label HCOLON max-chunk-size-value
max-chunk-size-label = "max-chunk-size"
max-chunk-size-value = 1*(DIGIT) ; chunk size in Kilobytes
```

Example of the “a=max-chunk-size” containing a MSRP chunk size of 1Mb:

```
               a=max-chunk-size: 1024
```

Appendix D. Release Version in User-agent and Server headers (Normative)

User-Agent and Server headers are used to indicate the release version and product information of the CPM Client, the CPM Participating Function and the CPM Controlling Function.

The CPM Client, the CPM Participating Function and the CPM Controlling Function shall implement the User-Agent and Server headers, according to the rules and procedures of [RFC3261] with the clarifications in this subclause specific for CPM.

The User-Agent and Server headers ABNF are specified in [RFC3261] and extended as follows:

```
Server = "Server" HCOLON server-val *(LWS server-val)
User-Agent = "User-Agent" HCOLON server-val *(LWS server-val)
server-val = product / comment
product = CPM-product / token [SLASH product-version]
product-version = token
```

D.1 CPM Version 1.0

This specification allows having several server-val tags. The first of those server-val tags shall be encoding according to the following ABNF:

```
CPM-product = "CPM-" cpm-device-token (SLASH cpm-product-version)
CPM-device-token = "client" | "serv" token
CPM-product-version = "OMA1.0"
```

Example 1:

In this example CPM Client acting as UAC and the CPM Participating Function acting as UAS are OMA CPM release version 1.0 products. The CPM Client has inserted its own company and product name and version "Arena-Messaging1000/v1.01".

User-Agent: CPM-client/OMA1.0 Arena-Messaging1000/v1.01

Server: CPM-serv/OMA1.0

Example 2:

In this example both the CPM Participating Function acting as UAC and the CPM Client acting as UAS are OMA CPM release version 1.0 products.

User-Agent: CPM-serv/OMA1.0

Server: CPM-client/OMA1.0

D.2 CPM Version 2.0

CPM Enabler version 2.0 uses the same server-val tags and encoding as defined by CPM version 1.0, where the ABNF encoding of the first of those server-val tags SHALL be according to following:

```
CPM-product = "CPM-" cpm-device-token (SLASH cpm-product-version)
CPM-device-token = "client" | "serv" token
CPM-product-version = "OMA2.0"
```

Example 1:

In this example CPM Client acting as UAC and the CPM Participating Function acting as UAS are OMA CPM release version 2.0 products. The CPM Client has inserted its own company and product name and version "Arena-Messaging1000/v1.01".

User-Agent: CPM-client/OMA2.0 Arena-Messaging1000/v1.01

Server: CPM-serv/OMA2.0

D.3 CPM Version 2.1

CPM Enabler version 2.1 uses the same server-val tags and encoding as defined by CPM version 2.0, with the following difference:

```
      CPM-product-version = "OMA2.1"
```

Appendix E. Examples of CPM-based Services

This appendix lists several examples of CPM-based Services. This list is not exhaustive and not intended to limit the development of further CPM-based Services.

E.1 Video Share Service along with a pre-existing Voice Call

The video share service along with pre-existing voice call allows a User to share a video continuous media stream with another user, along with a pre-existing voice-call (CS-voice or VOIP).

The CPM User can decide to share pre-recorded or live video with his contact.

The video share CPM-based Service uses the *+g.3gpp.cs-voice* feature tag to identify itself and the service is defined in [GSMA IR.74], and, with some extensions to [GSMA IR.74], in [GSMA IR 84].

The video share CPM-based Service uses the following facilities offered by CPM:

- Registration services as described in sections 7.1 "*Registering at the SIP/IP Core*" and 8.1 "*Registration*".
- CPM Session services as described in sections 7.3 "*CPM Session Handling*", 7.3.9 "*Media Plane Handling for CPM Sessions*", and 8.3.2 "*CPM Session Handling*", with the following restrictions:
    - The CPM Client sends a SIP INVITE request and establishes a half duplex RTP session as described in [GSMA IR.74] including CPM Conversation Identity as describe in 7.3.1.1 "*Initiating a CPM 1-1 Session*"
    - Only uni-directional video streams are supported.

A client uses SIP OPTIONS method to query video share service capability of terminating-party client(s), as described in [GSMA IR.74].

E.2 Video Share Service without a pre-existing Voice Call

The video share service without a pre-existing voice call allows a User to share a video continuous media stream with another user without a pre-existing voice-call.

The CPM User can decide to share pre-recorded or live video with his contact.

The video share CPM-based Service uses the ICSI(s) and IARI(s) defined in [GSMA IR 84] to identify itself and the service is defined in [GSMA IR 84].

The video share CPM-based Service uses the following facilities offered by CPM:

- Registration services as described in sections 7.1 "*Registering at the SIP/IP Core*" and 8.1 "*Registration*".
- CPM Session services as described in sections 7.3 "*CPM Session Handling*", 7.3.9 "*Media Plane Handling for CPM Sessions*", and 8.3.2 "*CPM Session Handling*", with the following restrictions:
    i. The CPM Client sends a SIP INVITE request and establishes a half duplex RTP session as described in [GSMA IR 84]including CPM Conversation Identity as describe in 7.3.1.1 "*Initiating a CPM 1-1 Session*"
    ii. Only uni-directional video streams are supported.

A client uses SIP OPTIONS method to query video share service capability of terminating-party client(s), as described in [GSMA IR 84].

Appendix F. The Parameters to be provisioned the CPM Service (Normative)

F.1 OMA CPM Device Management general

This appendix describes the parameters that are needed for initiation of the CPM service, as well as continuous provisioning by service providers. These parameters are specified in a Client Provisioning Application Characteristics document (AC file) and a Device Management Object (DM MO). The bootstrap function specified in [OMA Client Provisioning] and [OMA DM] SHALL be used to enforce the security of provisioning. Existing parameters in [OMA Provisioning Content] and [OMA DM] are re-used; those without corresponding parameters are defined in this specification and to be registered in OMNA through OMA official registration procedures.

1. APPID (Application ID): The application characteristics name for this application, to be used by the DM Client to uniquely identify the application.
2. NAME: Application name. To be displayed on the client side. This parameter is specific for each service provider.
3. PROVIDER-ID: provides an identifier for the application service access point.
4. TO-NAPID: allows an application to refer to a network access point with a matching NAPID parameter. It is only possible to refer to network access points defined within the same provisioning document.
5. TO-APPREF: parameter links the APPLICATION characteristics to another secondary APPLICATION characteristic with a matching APPREF parameter.
6. Max Ad-hoc Group size: Maximum number of Participants allowed for a CPM Group Session for a CPM Ad-hoc Group.
7. Controlling Function URI: A SIP URI used for setting up a CPM Group Session for a CPM Ad-hoc Group or for sending a Pager Mode or Large Message Mode CPM Standalone Message to a CPM Ad-hoc Group.
8. Message Storage Server URI: A URI for CPM user’s message store.
9. CPMDeferredMsgMgmtURI: A SIP URI for CPM user’s Deferred CPM Message function.
10. Max File Transfer Size: the maximum size in bytes per file for CPM File Transfer requests.

These parameters are to be registered in OMNA.

These parameters SHALL be obtained via the DM-1 reference point, from the data which is provisioned to the DM Client as specified in [OMA Provisioning Content] and [OMA DM].

Appendix G. Interoperability with OMA SIMPLE IM Clients

The CPM Participating Function MAY support interoperability with OMA SIMPLE IM clients, in order to support operators in deployments where an operator wants to support OMA SIMPLE IM clients with the CPM Enabler. The CPM Participating Function that supports interoperability with SIMPLE IM clients SHALL support only one type of clients per served User, i.e. either CPM clients or SIMPLE IM clients, but not a combination of them for the same user.

The CPM Participating Function MAY determine whether a served client is CPM or SIMPLE IM based on the registration information received according to section 8.1 "*Registration*".

In that case the CPM Participating Function SHALL act in the following manner:

A. When delivering a CPM request to an OMA SIMPLE IM Client, the terminating CPM Participating Function SHALL change in the CPM request the CPM Feature Tags so that it contains the IM Enabler identifier (i.e. +g.oma.sip-im), and for Large Message Mode CPM Standalone Messages, the +g.oma.sip-im.large-message feature tag. The CPM Participating Function SHALL NOT fetch the IM user preferences or apply any IM services to the incoming request, but shall only deliver the request towards the IM Client.

B. When receiving an OMA SIMPLE IM request, a CPM Participating Function SHALL change in the OMA SIMPLE IM request the IM Enabler identifier to the CPM Enabler identifier and appropriate CPM Enabler feature tags before routing the request. The CPM Participating Function SHALL also add Conversation-ID and Contribution-ID headers as defined in section 5.2.3.1 of [OMA-CPM-SD] (e.g. add a Conversation-ID with same value as the Contribution-ID). The value of the Conversation-ID will be set to the same value as the Contribution-ID, if received in the SIMPLE IM request.

C. Additional header fields and mappings are described in the table below.

NOTE 1: This means that OMA SIMPLE IM clients can receive CPM requests and that OMA SIMPLE IM clients can submit OMA SIMPLE IM requests towards the CPM Enabler.

NOTE 2: The above bullets require functionality in the SIP/IP core to send incoming CPM Standalone Messages, CPM Standalone Message disposition notifications, CPM File Transfers and CPM Sessions destined towards an OMA SIMPLE IM Client and incoming IM Messages, IM File Transfers and IM Sessions destined to a CPM Client to the CPM Participating Function.

NOTE 3: A network which deployed OMA SIMPLE IM only and which is to interoperate with CPM, is assumed to be able to deliver CPM requests to OMA SIMPLE IM Clients

OMA SIMPLE IM based services	Format and values of OMA IM Enabler Identifiers	Format and values of OMA CPM Enabler Identifier and Features Tags
SIMPLE IM Pager Mode, File Transfer, IM Session	In Contact and Accept-Contact headers: +g.oma.sip-im	In Contact and Accept-Contact headers: +g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.msg" or +g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.filetransfer" or +g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.session" In the P-Asserted-Service header: urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg or urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.group or urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer or urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer.group or urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session or urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group
SIMPLE IM Large Message Mode	In Contact and Accept-Contact headers: +g.oma.sip-im; +g.oma.sip-im.large-message	In Contact and Accept-Contact headers: +g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.largemsg" In the P-Asserted-Service header: urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg or urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group

Table 6: SIMPLE IM Enabler Feature Tags Mapping to/from OMA CPM Enabler Identifiers

Support for 1-1 Chat with SIMPLE IM Clients:

- When a 1-1 Session request is received from a OMA-SIMPLE-IM Client containing a first message in the SIP INVITE:

    i. the CPM Participating Function SHALL behave as a B2BUA and send a CPM 1-1 Session invitation to the recipient CPM Client and once the CPM Client accepted it, the CPM Participating Function provide the first message received in SIP INVITE from the OMA-SIMPLE-IM leg, into the MSRP session of the CPM leg. If the CPM 1-1 Session was not successfully set up, the CPM Participating Function MAY determine other ways to deliver the first message received in an 1-1 IM Session to the CPM Client, subject to service provider policies (e.g. sending the message as a CPM Standalone Pager Mode message if supported).

G.1 Media handling with SIMPLE IM Clients

If any MSRP interoperability is required, then proceed as described in section 5.2.1.4 “*Handling of Media connection parameters for MSRP session matching*“, otherwise continue with the following steps.

G.2 Handling Message Store

The CPM Participating Function MAY support storing in the CPM Message Store of messages, chat or File Transfer requests received from or for, the served SIMPLE IM users. In such case, the CPM Participating Function MAY store:

- sent and received 1-1 chat messages, 1-1 File Transfers, together with their associated IMDNs received or sent, in the corresponding 1-1 conversation folder based on the other party’s Authenticated Address, and
- sent and received Group chat messages, Group State Objects, Group File Transfers, together with their associated IMDNs received or sent, in the corresponding conversation folder based on their Contribution-ID and the assigned Conversation-ID.

Appendix H. CPM Feature Tags

This Appendix describes the CPM Feature identifiers used in this Technical Specification.

H.1 CPM Feature Identifiers

The CPM Feature identifiers allow the SIP/IP core to immediately recognize the enabler being invoked in order to route the request to the right application server. CPM Feature identifiers are carried in feature tags as described in section H.3 "*Proposed Formats for CPM Feature Identifiers*".

The CPM Enabler needs several identifiers in order to identify the signalling related to the CPM Enabler as it is processed by the CPM Client and the SIP/IP core as well as by the CPM Participating and Controlling Functions and to identify all the different features supported by the CPM Enabler.

The CPM Feature identifiers identify which CPM Feature the CPM Client is using or willing to use.

If the SIP/IP core corresponds to 3GPP/3GPP2 IMS, the CPM Feature identifiers are referred to as IMS Communication Service Identifiers, or ICSI.

When the 3GPP/3GPP2 IMS core receives the P-Preferred-Service header value, it verifies the value of this field and creates the P-Asserted-Service header value, as described in [3GPP TS 24.229] and deletes the P-Preferred-Service header value.

H.2 CPM Client Behaviour

The CPM Client SHALL identify all the CPM Features it supports and the CPM-based Services it supports by including the feature tags identifying them in the **Contact header** when it registers as described in [RFC3840].

It SHALL also include the feature tag being used for a particular feature in the **Contact header** when it issues any other SIP request except in the SIP MESSAGE request which does not contain a Contact header.

The CPM Client SHALL also identify the particular CPM Feature as well as any CPM-based Service it wishes to use during a particular communication by including the feature tags identifying them in the **Accept-Contact** header as described in [RFC3841].

The CPM Client SHALL identify the particular CPM Feature it is using when it issues specific SIP requests by including the particular CPM Feature in the **P-Preferred-Service** header as described in [3GPP TS 24.229].

H.3 Proposed Formats for CPM Feature Identifiers

When the SIP/IP core corresponds to 3GPP/3GPP2 IMS according to the rules and procedures of [3GPP TS 24.229]/[3GPP2 X.S0013.004], the CPM Feature identifier SHALL be set to:

1. +g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.<cpm-feature>" when it is carried as a feature tag in a Contact or Accept-Contact header; and,
2. urn:urn-7:3gpp-service.ims.icsi.oma.cpm.<cpm-feature> when it is carried as a URN in a P-Preferred-Service or P-Asserted-Service header as described in [3GPP TS 24.229].

When the SIP/IP core is not 3GPP/3GPP2, it is recommended that the same values of the CPM Features identifiers be used as when the SIP/IP core corresponds to 3GPP/3GPP2 IMS in order to facilitate interoperability.

The CPM Feature identifier carried as a feature tag SHALL be used by CPM Clients and CPM Participating and Controlling Functions, as specified in the procedures in this document.

The CPM Feature identifier carried as a URN in a P-Preferred-Service header SHALL be used by CPM Clients and carried as a URN in a P-Asserted-Service header SHALL be used by CPM Participating and Controlling Functions, as specified in the procedures in this document, when the SIP/IP core corresponds to 3GPP/3GPP2 IMS.

The <cpm-feature> value in the CPM Feature identifier further identifies the CPM Feature being invoked as described in Table 7 below.

CPM Features	Format and values for Accept-Contact and Contact	Format and values for P-Preferred-Service and P-Asserted-Service
Pager Mode CPM	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-	urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg

Standalone Message	service.ims.icsi.oma.cpm.msg"	OR urn:urn-7:3gpp-service.ims.icsi.oma.cpm.msg.group [see Note A below]
Large Message Mode CPM Standalone Message	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.largemsg"	urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg OR urn:urn-7:3gpp-service.ims.icsi.oma.cpm.largemsg.group [see Note A below]
CPM File Transfer	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.filetransfer"	urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer OR urn:urn-7:3gpp-service.ims.icsi.oma.cpm.filetransfer.group [see Note A below]
CPM Session	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.session"	urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session OR urn:urn-7:3gpp-service.ims.icsi.oma.cpm.session.group [see Note A below]
Deferred CPM Message	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.deferred"	urn:urn-7:3gpp-service.ims.icsi.oma.cpm.deferred
CPM Notification	+g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.oma.cpm.systemmsg”	urn:urn-7:3gpp-service.ims.icsi.oma.cpm.systemmsg

Table 7: Formats for CPM Feature identifiers

Note A: For Pager Mode CPM Standalone Message, Large Message Mode CPM Standalone Message, CPM File Transfer and CPM Session the P-Preferred-Service and P-Asserted-Service tag will support a “group” subclass that is usedfor Group CPM communications (i.e. Group CPM Standalone Message, Group CPM Session, Group CPM File Transfer).

H.4 Client CPM-based Service Identification

A CPM-based Service SHALL be identified by its own feature tag. If the SIP/IP core corresponds to 3GPP/3GPP2 IMS, the IMS Application Reference Identifier, or IARI is used to carry this CPM-based Service identifier in the Accept-Contact header according to the rules and procedures of [3GPP TS 24.229]/[3GPP2 X.S0013.004]

CPM-based Service	**Format and values for Accept-Contact and Contact**
CPM-based Service	+g.3gpp.iari-ref="urn%3Aurn-7%3A3gpp-application.ims.iari.oma.cpm.<*iari-value*>"

Table 8: CPM-based Service identifiers

H.5 P-Asserted-Service Header Field

A CPM-based Service SHALL be identified by its own ICSI and based on [3GPP TS24.229] an Application Server (i.e. CPM Participating Function) MAY populate the P-Asserted-Service header field with the appropriate asserted ICSI for the service, if the S-CSCF has not done it hence the P-Preferred-Service header field is received in the request by the CPM Participating Function.

For CPM requests, the terminating SIP/IP Core network elements SHOULD NOT remove the P-Asserted-Service header field when delivering to the terminating CPM Clients.

Appendix I. CPM Notification Formats

I.1 Deferred CPM Message Notification Format

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

The purpose of the out-of-band notification is to inform the CPM Client that one or more Deferred CPM Messages are stored in the CPM Participating Function. Based on this information, the CPM Client or the CPM User MAY decide to register for the CPM service in order to retrieve the deferred messages.

The out-of-band Deferred CPM Message Notification is sent using the Push enabler. The CPM Participating Function SHALL submit a Push message as described in section 5.1 of [OMA-PAP]. The message SHALL contain:

- A control entity, as defined in [OMA-PAP] section 8.2, and
- A content entity carrying metadata about the deferred CPM Message formatted according to the "application/msginfo+xml" MIME type defined in Appendix J.1 "*Deferred Messages Metadata*". When the notification is to be sent via an SMS, the CPM Participating Function SHALL ensure that the size of the content entity is at most 140 bytes.

The Push message SHALL be formatted according to [OMA-Push-Message] and the parameters of the message are described in [OMA-Push-OTA].

NOTE: a new application identifier "X-Wap-Application-Id" must be assigned by and registered with OMNA for identifying the CPM enabler.

I.1.2 In-band Deferred CPM Message Notification Format

The in-band Deferred CPM Message notification is sent in a SIP MESSAGE request as specified in section 8.3.1.6.3 "*Sending Notifications and Awaiting CPM Client Action*".

The body of the message SHALL be an XML document carrying metadata about the Deferred CPM Message.

The metadata is the same as the metadata defined for the <deferred-messages> event package in Appendix J.1 "*Deferred Messages Metadata*" with the stipulation that only one <message> element is included.

I.2 Notification Format of Deferred CPM Message After Expiry

The purpose of this notification is to let the CPM Client know how a Deferred CPM Message was handled upon expiry. This notification is handled as described in section 7.2.3.3 "*Receiving a notification for Expiry of Deferred CPM Standalone Message".*

I.2.1 Out-of-band Notification

The out-of-band notification of Deferred CPM Message after expiry is sent using the Push enabler.

The CPM Participating Function SHALL submit a Push message as described in section 5.1 of [OMA-PAP]. The message SHALL contain:

- a control entity, as defined in [OMA-PAP] section 8.2, and,
- a content entity carrying information about the deferred CPM Message after expiry, formatted according to the "application/msginfo+xml" MIME type defined in Appendix K "*Format of Notification for Deferred CPM Message after Expiry*".

The Push message SHALL be formatted according to [OMA-Push-Message] and the parameters of the message are described in [OMA-Push-OTA].

I.2.2 In-band notification

The in -band Deferred CPM Message Notification is sent in a SIP MESSAGE request as specified in section 8.3.1.6.8 "*Handling Deferred CPM Messages on Expiry Time*".

The body of the message SHALL be an XML document carrying information defined in Appendix K "*Format of Notification for Deferred CPM Message after Expiry*".

Appendix J. Deferred Messages event package definition

A new event package is defined for deferred messages. The CPM Client subscribes to this event to receive information about deferred messages at registration. The CPM Participating Function sends metadata about the deferred messages using this new event package.

NOTE: This event package needs to be registered with IANA and a new draft will be written and presented to IETF.

This section fills in the information required for all event packages by [RFC6665].

- Package Name

  The name of this package is "deferred-messages”

As specified in [RFC6665], this value appears in the Event header field present in SUBSCRIBE and NOTIFY requests.

Example:

Event: deferred-messages

- Event Package Parameters

  The SIP event framework allows event packages to define additional parameters carried in the Event header field. This package, deferred-messages, does not define any additional parameters.

- SUBSCRIBE Bodies

  There is no body for the SUBSCRIBE to this event package

- Subscription duration

  Only subscriptions with an Expires header of zero “0” are supported;

- NOTIFY body

  The body of the NOTIFY will consist of an XML body containing the deferred messages metadata as described below.

The namespace URI for elements defined in this proposed event package is a URN, using the namespace identifier 'ietf'. This URN is:

urn:ietf:params:xml:ns:msginfo

A new MIME type, application/msginfo+xml is defined for this event package:

MIME media type name: application

MIME subtype name: msginfo+xml

J.1 Deferred Messages Metadata

J.1.1 Structure

The root element of the Deferred Messages Metadata document is a <message-list> element.

The <message-list> element:

1. MAY contain a “number” attribute indicating the number of deferred messages currently stored in the CPM Participating Function;
2. MAY contain one or more <message> elements containing descriptive information regarding each deferred message;
3. MAY include any other attributes from any other namespaces for the purposes of extensibility.

The <message> element:

1. MAY include a "date-time" attribute representing the date and time when the message was stored by the Deferred Function.
2. MAY include a "message-reference" attribute representing the complete path that uniquely identifies the message;

NOTE: The “message-reference” attribute is expected to be used only within in-band notifications.

3. MAY include a <size> element representing the size of the stored content;
4. MAY include an <expiry> element representing the date at which the message expires;
5. MAY include a <subject> element representing the Subject header field of the SIP request;
6. MAY include a <UPP-ID-list> element indicating the list of identifiers of User Preference Profiles of the CPM User for which no rule containing the action element <allow-reject-invite> with the value set to “true” is matching for the deferred message;

   NOTE: The content of this element can be used by the CPM Client to decide how to handle the notification by comparing it with his active User Preference Profile.

7. MAY include an <info> element, containing:
   a. a mandatory <from> element taken from the “From” header field of the SIP request;
   b. optionally, one of the following elements, indicating the recipient(s) of the message
      i. the <to> element containing the “To” header field of the SIP request if the message is sent to one CPM User; or
      ii. the <recipient-list> element if the message is aimed to multiple users. This element contains a copy of intended recipients, e.g. the group session participants or the URIs listed in the URI-list case as described in [RFC5365] or [RFC5368] ;
   c. an optional <auth-id> element representing the Authenticated Originator’s CPMAddress as defined in section 6.1 “*Authenticated Originator’s CPM Address*” if Privacy setting allows;
   d. optionally any other attributes from any other namespaces for the purposes of extensibility.
8. MAY include any other attributes from any other namespaces for the purposes of extensibility;

The <recipient-list> element SHALL contain one or more <entry> elements. Each <entry> element:

1. SHALL include a "uri" attribute representing the URI of a recipient for the message;
2. MAY include a <display-name> element, containing a human-readable name corresponding to the message recipient.

In case of pre-defined groups, the <recipient-list> element SHALL contain the “group-uri” and MAY contain the “group-display-name” attributes.

Appendix K. Format of Notification for Deferred CPM Message after Expiry

The root element is <message-expired> that includes the attributes below.

1. SHALL include a "message-reference" attribute representing the complete path that uniquely identifies the message (i.e. message-URI-ID);
2. If the Deferred CPM Message was stored to Message Storage Server on expiry, SHALL include a “UID” attribute set to the UID value retrieved from the Message Storage Server.

Appendix L. Emoticons (Normative)

L.1 List of supported emoticons

The CPM enabler allows a client to take advantage of emoticons. A CPM Client taking advantage of such feature will replace specific character sequences known as emoticons when a CPM Message is received and is displayed to the CPM User, or when the CPM User composes a CPM Message the emoticons are replaced with the specific character sequence(s) before the CPM Message is sent. The exact behaviour for CPM Clients is described in L.2. This feature is an implementation choice, and as such is OPTIONAL for the client. The list of character sequences and emoticons are listed in Table below. The table captures only a subset of emoticons that are widely used - client implementations MAY extend the list with additional character sequences. The first column in the table identifies the emoticons; the second column defines one or more character sequences for each emoticon so that different client implementations can associate the same meaning to a character sequence and ensure interoperability between them. The third column provides a description of possible graphical renditions. Note that the third column provides examples only and client implementations are not constrained to follow the examples. The exact images representing the individual emoticons are not in the scope of this enabler – it is a client implementation choice.

Emoticons	Character sequences	Examples describing graphical renditions
Happy, smile	:-) or :)	A happy or smiling face
Sad	:-(or :(	A sad face
Wink	;-) or ;) or ;o) or ;O)	A winking face
Big grin	:-D or :D or :oD or :-d or :d or :od or :Od or :OD	A big grin face
Confused	:-/ or :-\	A confused face
Blushing, embarrassed	:”-) or :”) or :”> or :-$ or :$	A blushing, embarrassed face
Stick-out tongue	:-P or :P or :oP or :-p or :p or :op or :OP or :Op	A stick-out tongue face
Kiss, red lips	:-* or :*	A kissing face or red lips
Shocked, surprised	:-O or :-o or :o or :O	A shocked, Surprised face
Angry	:-@ or :@ or X-(or X(or x-(or x(or xo(or XO(	An angry face
Cool, sunglasses	B) or B-) or (H) or (h) or Bo) or BO)	A face with sunglasses
Worried	:-S or :S or :-s or :s or :oS	A worried face
Devilish	>:-) or >:) or >:o) or >:O)	A devilish face
Crying	:,-(or :,(or :’-(or :’(or :,o(or :’o(or :,O(or :’O(	A crying face
Laughing	:-)) or :)) or :o)) or :O))	A laughing face
Straight face, disappointed	:-| or :| or :o| or :O|	A straight face
Angel, innocent	O:-) or O:) or o:-) or o:)	An innocent face
Nerd	:-B or :B	A nerdish face
Sleepy	|-O or |O or |-o or |o	A sleepy face
Rolling eyes	8-) or 8) or 8o) or 8O)	A rolling eyes face
Sick, berk	:-& or :& or ;o& or :O&	A sick/ill face
Shhh! No speak, lips sealed	:-SS or :SS or :ss or :-ss	A face with sealed lips
Thinking, pensive	:-? or :?	A pensive face

Emoticons	Character sequences	Examples describing graphical renditions
Raised eyebrow, sarcastic look	/:-) or /:) or /:o) or /:O)	A raised eyebrow face or a face with a sarcastic look
Rose, flower	@):-	A rose
Cup of coffee	~o)	A cup of coffee
Drink, cocktail	)-|	A cocktail glass
Idea (light bulb)	*-:-) or *-:)	A light bulb
Love struck, heart	(L) or <3	A heart

Table 9: Character sequences and emoticons

L.2 Emoticon processing by the CPM Clients

The CPM Client will process the character sequences associated with each emoticon, converting the character sequences from/to their corresponding image representation when an CPM Message is received/displayed or composed/sent.

L.2.1 Emoticons in composed/sent instant messages

Whenever a CPM Message is composed by the CPM User, the CPM Client MAY replace the recognized character sequences with the corresponding emoticon as the user types them into the CPM Message. Implementations of CPM Client MAY also offer the possibility to insert the emoticons using shortcuts. When the CPM Client uses shortcuts to insert emoticons into the CPM Message, it SHOULD choose the first character sequence from table above [Table 9] corresponding to the inserted emoticon. In any case, when the CPM Client sends the CPM Message, the recognized emoticons SHALL be converted to their corresponding character sequence before the CPM Client sends the CPM Message to the CPM Participating Function.

L.2.2 Emoticons in received/displayed instant messages

Whenever a CPM Message is received or displayed by/for the user, the CPM Client MAY replace the recognized character sequences with the corresponding emoticon.

Appendix M. Example of Event Reporting body (Normative)

Examples of the data structure of the event reporting framework structure follows:

```
            <!--  example 1: set \Seen flag on group chat object stored in /Default folder  -->
<?xml version="1.0" encoding="UTF-8"?>
<cpm-evfw xmlns="urn:oma:xml:cpm:evfw" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

          <event-cpm-imap  type="flags">
               <operation name="add " flag="\Seen">
                      <!--   indicates the message object on which the operation should be performed -->
                      <!--  uid attribute is optional, shall be populated if known by the CPM Client -->
                      <!--  folder-path attribute is optional, shall be populated if known by the CPM client -->

                      <object uid=58175 folder-path="Default/thisConversationID1234/thisContributionID5678/">
                              <message-id>7QsgkAl8QZvr9XPFtiiCtLj3sls0oPof</message-id>
                              <!--  mandatory element, this is the IMDN message-ID element, syntax defined in RFC5438
                              section. 6.3 -->

                              <extension-elements-per-object-here> <!-- EXTENSION POINTS, ADD HERE FUTURE NEW
                              ELEMENTS -->
                              </extension-elements-per-object-here>
                      </object>
               </operation>
          </event-cpm-imap >
</cpm-evfw>

_________________________________________________________________________________________
           <!--  example 2: set \Seen flag on a 1-1 chat object for which CMS synch was not yet done  -->
<?xml version="1.0" encoding="UTF-8"?>
<cpm-evfw xmlns="urn:oma:xml:cpm:evfw" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

          <event-cpm-imap  type="flags">
               <operation name="add " flag="\Seen">
                     <object>
                              <conversation-id>ruhfdui134t9187698t1rgf</conversation-id>
                              <contribution-id>uhfdui134t9187698t1rgfr</contribution-id>
                              <other-party>tel:+15145554321</other-party>    <!--  optional element -->
                              <message-id>7QsgkAl8QZvr9XPFtiiCtLj3sls0oNok</message-id>

                              <extension-elements-per-object-here> <!-- EXTENSION POINTS, ADD HERE FUTURE NEW
                              ELEMENTS -->
                              </extension-elements-per-object-here>

                      </object>
               </operation>
          </event-cpm-imap >
</cpm-evfw>

_________________________________________________________________________________________
            <!--  example 3: set \Deleted flag on group chat object stored in /UserFolder1 folder  -->
 <?xml version="1.0" encoding="UTF-8"?>
<cpm-evfw xmlns="urn:oma:xml:cpm:evfw" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

          <event-cpm-imap  type="flags">
             <operation name="add" flag="\Deleted">
                      <object uid=58177 folder-path="UserFolder1/thisConversationID1234/thisContributionID5678/">
                              <message-id>7QsgkAl8QZvr9XPFtiiCtLj3sls0765</message-id>
                      </object>
               </operation>

               <extension-elements-here>  <!--     EXTENSION POINTS, ADD HERE NEW ELEMENTS FOR IMAP
               EVENTS, IF NEEDED -->
```

```
                </extension-elements-here>

          </event-cpm-imap >
</cpm-evfw>
```

Table 10: CPM Reporting Framework example

Appendix N. Algorithm for Correlating SMS messages with Objects from the CPM Message Store (Normative)

The CPM Participating Function storing a message that was successfully delivered via SMS, SHALL store the prefix of the SMS text body (as defined in N.1) in the Message-Correlator MIME header of the message object.
The CPM Client SHALL use this Message-Correlator header value, along with “To”/”From” headers to find the corresponding locally stored SMS message.

N.1 Correlation algorithm

Editor’s Note: The algorithm used in this Appendix is ported from RCC.07 v6.0 section 4.1.4.4 and 4.1.4.5., hence this section is pending GSMA feedback on copyright. If any issues the text in N.1 and N.2 shall be removed.

The correlation is based on the following field values of the SMS message:

1) “To” header:
    - It SHOULD be the format as taken from the address field defined in [3GPP TS 23.040]. If TON (Type Of Number) indicates “international”, then a “+” is inserted before the number string. If TON indicates “unknown” only the number string is used. If the address is “alphanumeric”, then the address SHALL be encoded to UTF-8 format.
2) “From” header:
    - It SHOULD be the format as taken from the address field defined in [3GPP TS 23.040]. If TON indicates “international”, then a “+” is inserted before the number string. If TON indicates “unknown” only the number string is used. If the address is “alphanumeric”, then the address SHALL be encoded to UTF-8 format.
3) The Message-Correlator header value is generated as defined in Appendix F of [OMA-CPM_TS_IWF].

Additional considerations:

- For the correlation of outgoing messages the From field is not used
- For the correlation of incoming messages the To field is not use
- The correlation is achieved by Message-Correlator header value, using a case-sensitive comparison.
  The matching algorithm should take into account differences in the presentation of the address string according to different types of numbers.
  The creation of a Message-Correlator header value used for the correlation via a full string match requires in some scenarios access to the native SMS Transfer Protocol Data Units (TPDU, i.e. the TP-UD, TP-DCS data units). Client implementations that do not have access to the TPDU but only to the "interpreted" payload of the short message or if the message did contain characters encoded via single or locking shift tables may compensate this by alternative matching algorithms, e.g. a pattern match algorithm based on a smaller sequence of characters taken from the message content. Such a search algorithm increases the probability of collisions or matching errors, but may be compensated by multiple probes with different sequences of characters taken from the message content with a statistical evaluation.

N.2 Dealing with Collisions

The correlation field values SHALL be used to correlate between message objects stored with legacy MIME headers in the Message Store Server and the SMS messages received on the CPM Client. Specifically, when the CPM Client synchronizes with the Message Store Server it SHALL obtain UIDs and the correlation field values for those SMS messages that are new or have changed since the last synchronization. The CPM Client SHALL then attempt to correlate the UIDs and correlation field values of the message obejcts retrieved from the Message Store Server with any SMS messages it has received, or it subsequently receives, from the network. Therefore, if any of the messages have the same correlation field values (this is considered a correlation “collision”) then the CPM Client cannot distinguish between them when matching to its local messages.

The CPM Client SHOULD compare the direction (originating or terminating) in addition to comparing the correlation field values, meaning that correlation collisions can only occur on messages with the same direction.

Correlation collisions can occur in these two cases:

1. Messages in the same thread with the same content, typically when they are chronologically close (so returned on the same synchronization) SMS messages in the same thread with the same content, such as successive replies both saying “OK”.
2. Messages in the same thread with content that is different only after the first 140 bytes. This is more likely when higher numbers of messages are being compared, for example, a likely worst case example would be when a phone has been switched off for a long period (e.g. a vacation, a repair). This rare scenario is not addressed here further.

If there are collisions, the CPM Client should identify the chronologically first received message on the device, which has the lower UID on Message Store Server.

For example, suppose Message Store Server returns two new messages both with the same value C for the Message-Correlator header but with UIDs x and y, x < y, and the CPM Client has received two messages with the same value C for the Message-Correlator header at times t1 and t2, t1 < t2. Then the CPM Client should identify t1 = x and t2 = y.

The same principle applies when the number of correlation collisions on the device is different from the number on the Message Store Server; those are usually cases of temporary lack of synchronization between the CPM Client and the Message Store Server.

As an example, suppose as above the Message Store Server has the same two new messages but the CPM Client has only received one message with value C for the Message-Correlator header. It should identify that with UID x, in the presumption that the network will shortly deliver a second message with value C for the Message-Correlator header which it will then identify with UID y. Similarly, if the Message Store Server only has UID x producing value C for the Message-Correlator header but the CPM Client has both t1 and t2, the CPM Client should identify t1 with message x and expect a subsequent synchronization to return message y which it will then identify with t2.

The length of time between the messages SHOULD be considered by the CPM Client when determining whether the messages are duplicates. Note also that the CPM Client SHOULD have to take into account messages the device might have, that it received before the Message Store Server was in place.

The impact of correlation collisions in this method may result in a wrong correlation; in the case above, to identify t1 = x and t2 = y when the correct mapping was in fact t1 = y and t2 = x. In this case, the view from one device and another will be out of sync: a user making a state change to t1 on one device will see it applied to t2 on the other device, when they would expect it to apply to message y.

For example, take the case of successive identical messages. If the CPM User marks on one device the earlier of these messages as a favourite, then the device view might be as follows:

“are you still on for tonight?”
“yes” <- FAVOURITE
“do you have the tickets?”
“yes”

whereas on another device the view would be:

“are you still on for tonight?”
“yes”
“do you have the tickets?”
“yes” <- FAVOURITE

Appendix O. CPM Extensions for IMDN (Normative)

New elements are defined as extensions to the XML namespace defined in [RFC5438] (and [RFC5438Errata]), under the <status> element, to cover the scenarios of delivery performed outside the IMS domain, based on the schema extensibility rules defined in section 11.1.9 of [RFC5438].

The CPM IMDN extensions namespace is: "urn:oma:xml:cpm:imdn-extensions:1.0".

A new notification type is defined: <interworking-notification> element with the following possible child elements:

a) <legacy-sms> to indicate a successful delivery via SMS; or

b) <legacy-mms> to indicate a successful delivery via MMS; or

c) <error> to indicate a failed interworking delivery attempt.

The relaxNG schema for this extension is defined in Appendix O.2.

O.1 ABNF Update

The CPM extension updates the section 10 of [RFC54328] to add the value “interworking”, as follows:

```
Disposition-Notification =
       "Disposition-Notification" ": "
       [(notify-req *(COMMA notify-req))]
   notify-req =
       ("negative-delivery" / "positive-delivery" /
        "processing" / "display" /"interworking"/  Token) *(SEMI disp-notify-params)
   disp-notify-params = Ext-param
   Message-ID = "Message-ID" ": " Token
   Original-To = "Original-To" ": "  [ Formal-name ] "<" URI ">"
   IMDN-Record-Route =
       "IMDN-Record-Route" ": "  [ Formal-name ] "<" URI ">"
   IMDN-Route = "IMDN-Route" ": "  [ Formal-name ] "<" URI ">"
   SEMI    =  *SP ";" *SP ; semicolon
   COMMA   =  *SP "," *SP ; comma
```

O.2 The RelaxNG Schema

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

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="urn:oma:xml:cpm:imdn:1.0"
  xmlns:imdn="urn:ietf:params:xml:ns:imdn"
  xmlns="urn:oma:xml:cpm:imdn-extensions:1.0" version="1.0
 elementFormDefault="qualified">

  <xs:annotation>
    <xs:documentation>This schema defines the XML data types for the CPM IMDN extensions to the RFC5438
namespace</xs:documentation>
  </xs:annotation>

  <xs:import namespace="urn:ietf:params:xml:ns:imdn"
  schemaLocation="https://www.iana.org/assignments/xml-registry/schema/imdn.xsd"/>

  <!-- ========================================================== -->
  <!--    CPM extensions to IMDN - Data type definitions for "deliveryExtension" -->
  <!-- ========================================================== -->

 <!-- interworking-notification <status> CPM extension point -->
  <imdn:ref name="iwNotification"/>

 <xs:define name="iwNotification">
      <imdn:element name="interworking-notification">
            <imdn:element name="status">
                  <xs:choice>
                        <xs:element name="legacy-sms">
                              <empty/>
                        </xs:element>
                        <xs:element name="legacy-mms">
                              <empty/>
                        </xs:element>
                        <imdn:ref name="commonDispositionStatus"/>
                   </xs:choice>
                   <imdn:ref name="iwExtension"/>
            </imdn:element>
      </imdn:element>
</imdn:define>

</xs:schema>
```

Appendix P. The CPM Group Session Data Management (Normative)

P.1 Schema Description

The following describes the XML schema used for CPM Group Session Data Management defined in section 6.8 “*CPM Group Session Data Management*”:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:oma:xml:cpm:groupdata:1.0 "
xmlns=" urn:oma:xml:cpm:groupdata:1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified"
attributeFormDefault="unqualified">

<xs:element name="cpm-group-management">
  <xs:complexType>
      <xs:sequence>
          <xs:attriute name="id" type="integer" use="required"/>
          <xs:element name="group-data" type="group-data-type" minOccurs="0"/>
          <xs:any minOccurs="0" maxOccurs="unbounded" processContents="lax"/>
      </xs:sequence>
      <xs:anyAttribute namespace="##other" processContents="lax"/>
    </xs:complexType>
</xs:element>

<xs:element name="group-data-type">
    <xs:complexType>
       <xs:choice>
             <xs:element name="request" type="request-type"/>
             <xs:element name="response" type="response-type"/>
             <xs:any minOccurs="0" maxOccurs="unbounded" processContents="lax"/>
        </xs:choice>
        <xs:anyAttribute namespace="##other" processContents="lax"/>
     </xs:complexType>
</xs:element>

<xs:element name="request-type">
    <xs:complexType>
       <xs:sequence>
           <xs:attribute name="target" type=" type="target-type">
           <xs:element name="action" type="action-type">
           <xs:element name="data" type="data-type" minoccurs="0">
           <xs:any minOccurs="0" maxOccurs="unbounded" processContents="lax"/>
        </xs:sequence>
        <xs:anyAttribute namespace="##other" processContents="lax"/>
      </xs:complexType>
</xs:element>

<xs:element name="response-type">
      <xs:complexType>
           <xs:sequence>
                 <xs:element name="response-code" type="xs:integer" use="required"/>
                 <xs:element name="response-text" type="xs:string" minOccurs="0">
                 <xs:any minOccurs="0" maxOccurs="unbounded" processContents="lax"/>
            </xs:sequence>
            <xs:anyAttribute namespace="##other" processContents="lax"/>
       </xs:complexType>
</xs:element>

<xs:element>
     <xs:simpleType name="target-type">
         <xs:restriction base="xs:string">
              <xs:enumeration value="subject"/>
              <xs:enumeration value="icon"/>
             <xs:enumeration value="role"/>
          </xs:restriction>
      </xs:simpleType>
</xs:element>
```

```
<xs:element>
      <xs:simpleType name="action-type">
            <xs:restriction base="xs:string">
                     <xs:enumeration value="set"/>
                     <xs:enumeration value="delete"/>
                     <xs:enumeration value="move"/>

            </xs:restriction>
       </xs:simpleType>
</xs:element>

<xs:element name="data-type">
     <xs:complexType>
            <xs:choice>
                     <xs:element name="subject" type="xs:string"/>
                     <xs:element name="icon" type="xs:icon-type"/>
                     <xs:element name="bulletin" type="xs:bulletin-type"/>
                     <xs:element name="role" type="xs:role-type"/>

                     <xs:any minOccurs="0" maxOccurs="unbounded" processContents="lax"/>
             </xs:choice>
             <xs:anyAttribute namespace="##other" processContents="lax"/>
        </xs:complexType>
</xs:element>

<xs:element name="icon-type">
     <xs:complexType>
            <xs:choice>
                    <xs:element name="icon-uri" type="xs:anyURI"/>
                    <xs:element name="file-info" type="xs:string"/>
                    <xs:any minOccurs="0" maxOccurs="unbounded" processContents="lax"/>
            </xs:choice>
            <xs:anyAttribute namespace="##other" processContents="lax"/>
      </xs:complexType>
</xs:element>

<xs:element name="bulletin-type">
          <xs:complexType>
               <xs:sequence>
                    <xs:element name="bulletin" type="xs:string"/>
                    <xs:element name="participant" type="participant-type" minOccurs="0"/>
                    <xs:element name="timestamp" type="dateTime" minOccurs="0"/>
                    <xs:any minOccurs="0" maxOccurs="unbounded" processContents="lax"/>
               </xs:sequence>
               <xs:anyAttribute namespace="##other" processContents="lax"/>
          </xs:complexType>
</xs:element>

<xs:element name="role-type">
     <xs:complexType>
            <xs:choice>
                    <xs:element name="user-role" type="xs:string"/>
                    <xs:element name="participant" type="xs:anyURI"/>
                    <xs:any minOccurs="0" maxOccurs="unbounded" processContents="lax"/>
            </xs:choice>
            <xs:anyAttribute namespace="##other" processContents="lax"/>
      </xs:complexType>
</xs:element>

</xs:schema>
```

Table 11: CPM Group Session Data Management Schema

P.2 CPM Group Session Data Management Examples

An example of a CPM Group Session Data Management request to set a subject of a CPM Group Session is provided in the table below.

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

<cpm-group-management
  xmlns="urn:oma:xml:cpm:groupdata:1.0"
  id="7598235857">
  <group-data>
    <request target="subject">
      <action>set</action>
      <data>
        <subject>Hello World</subject>
      </data>
    </request>
  </group-data>
</cpm-group-management>
```

Table 12: CPM Group Session Data Management Request Set Example

An example of a CPM Group Session Data Management request to delete an icon of a CPM Group Session is provided in the table below.

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

<cpm-group-management
  xmlns="urn:oma:xml:cpm:groupdata:1.0"
  id="iuhghuwerpghaposif">
  <group-data>
    <request target="icon">
      <action>delete</action>
    </request>
  </group-data>
</cpm-group-management>
```

Table 13: CPM Group Session Data Management Request Delete Example

An example of a CPM Group Session Data Management request to move the CPM User's administrator role to another CPM Group Session Participant is provided in the table below.

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

<cpm-group-management
  xmlns="urn:oma:xml:cpm:groupdata:1.0"
  id="5795985717498712987547">
  <group-data>
    <request target="role">
      <action>move</action>
      <data>
        <user-role>Administrator</user-role>
        <participant>tel:+491711234567</participant>
      </data>
    </request>
  </group-data>
</cpm-group-management>
```

Table 14: CPM Group Session Data Management Request Move Example

An example of a CPM Group Session Data Management success response to the request shown in Table 11 is provided in the table below.

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

<cpm-group-management
  xmlns="urn:oma:xml:cpm:groupdata:1.0"
  id="7598235857">
  <group-data>
    <response>
      <response-code>200</response-code>
      <response-text>OK</response-text>
    </response>
  </group-data>
</cpm-group-management>
```

Table 15: CPM Group Session Data Management Response Example

P.3 CPM Extensions to the Conference Event Package

The following describes the XML schema that extends the current XML schema specified in [RFC 4575] to add the "icon" and "subject-ext" XML elements.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:oma:xml:cpm:conference-info:1.0 "
xmlns=" urn:oma:xml:cpm:conference-info:1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified"
attributeFormDefault="unqualified">

<xs:annotation>
    <xs:documentation>
       Adds elements to the conference description of the conference event package to convey icon and subject information.
   </xs:documentation>
</xs:annotation>

<xs:import namespace="urn:ietf:params:xml:ns:conference-info"
schemaLocation="urn:ietf:params:xml:schema:conference-info"/>

<xs:element name="icon">
    <xs:complexType>
       <xs:sequence>
          <xs:element name="source" type="source-type" minOccurs="0"/>
          <xs:element name="participant" type="xs:anyURI" minOccurs="0"/>
          <xs:element name="timestamp" type="xs:dateTime" minOccurs="0"/>
          <xs:any minOccurs="0" maxOccurs="unbounded" processContents="lax"/>
       </xs:sequence>
</xs:complexType>
 </xs:element>

<xs:element name="subject-ext"/>
      <xs:complexType>
             <xs:sequence>
                        <xs:element name="participant" type="xs:anyURI"/>
                        <xs:element name="timestamp" type="xs:dateTime"/>
                        <xs:any minOccurs="0" maxOccurs="unbounded" processContents="lax"/>
              </xs:sequence>
     </xs:complexType>
     <xs:complexType name="source-type">
             <xs:choice>
   <xs:element name="icon-uri" type="xs:anyURI"/>
   <xs:element name="file-info" type="xs:string"/>
   <xs:any minOccurs="0" maxOccurs="unbounded" processContents="lax"/>
              </xs:choice>
     </xs:complexType>
</xs:element>

</xs:schema>
```

Table 16: Conference Event Package CPM Extensions

The elements "icon" and subject-ext" are an extension of the <conference-description> element in the conference XML document of [RFC 4575]. Therefore, these elements SHALL NOT be included in any other element than <conference-description>.

The "icon" element is used to convey to the CPM Client information related to the icon assigned to a CPM Group Session.

The "subject-ext" element is used to convey to the CPM Client additional information related to the "subject" of the CPM Group Session.

Appendix Q. CPM Ad-hoc Group Policy (Normative)

This Appendix defines mechanisms for:

- communicating policies that are applied by that CPM Controlling Function for CPM Group Sessions, and
- the assignment of roles to Participant CPM Users being of CPM Group Sessions.

Q.1 CPM Controlling Function Policy

The CPM Controlling Function MAY communicate to the Participant CPM Clients the policies in operation for the Participants of the CPM Group Sessions.

The following policies of the CPM Controlling Function MAY be indicated:

- Participant Removal Policy.

Q.1.1 Participant Removal Policy

The Participant Removal Policy indicates the policy applied by the CPM Controlling Function for Participants to request the removal of another Participant, as defined in sections 7.3.6 "*Remove Participants from a CPM Group Session*" and 9.2.6 "*Removing Participant Request*".

The following Participant Removal Policies are supported:

1) all Participants:
All CPM Users participating in a CPM Group Session are allowed to request removal of a Participant. This is the default value, if no Participant removal policy is indicated.

2) Administrator only:
Only CPM Users participating in a CPM Group Session with an assigned administrator role are allowed to request removal of another Participant. CPM Client(s) of the Participants that do not have the administrator role SHALL not offer to the CPM User the option to remove Participants.

3) no Participant Removal:
No CPM User participating in a CPM Group Session is allowed to request removal of another Participant. The CPM Client(s) of Participants SHALL not offer to the CPM User to the option to remove another Participant.

   This SHALL be assumed to be the applicable policy unless indicated otherwise.

Q.1.2 User Role Assignment Policy

The User Role Assignment Policy indicates the assignment policy applied by the CPM Controlling Function for User Roles in a CPM Group Session.

The following User Role Assignment Policies are supported:

1) Single Administrator:
Zero or one Participants of the CPM Group Session can have the administrator role assigned. Only a CPM Client with a administrator role can move its role to another participant in the CPM Group Session. CPM Clients without administrator role are not entitled to perform CPM Group Session Data Management request with a <target> element indicating "role".

2) Any Administrator:
Any Participant in the CPM Group Session can be assigned with the administrator role. CPM Clients assigned with the Administrator role can set or delete any User Role of any other Participant in the CPM Group Session or can move their administrator role to another Participant via a CPM Group Session Data Management request as defined in section 6.8 *"CPM Group Session Data Management"*. CPM Clients without administrator role are not entitled to perform CPM Group Session Data Management request with a <target> element indicating "role".

If the User Role Assignment Policy is absent or if none of the Participants in a CPM Group Session is assigned with the Administrator role as defined in Appendix Q.1.2 *"CPM User Roles"*, then no CPM Client is entitled to perform CPM Group Session Data Management request with a <target> element indicating "role".

Q.1.3 User Role Value Encoding

The supported policies are indicated via CPM well-defined keywords in the <keywords> element of the <conference-description> element of the conference event package as defined in [RFC4575].

The value of the <keywords> element of the conference event package of the CPM Controlling Function SHALL conform the definitions of the xml schema list element following the ABNF syntax:

```
keywords-value = keyword *(%x20 keyword)

keyword = well-known-keyword / token

well-known-keyword = removal-policy | admin-policy

removal-policy = rem-all / rem-admin / rem-no

rem-all = "rem-all"; indicates the "all Participants" policy

rem-admin = "rem-administrator" ; indicates the "Administrator only" policy

rem-no = "rem-nobody" ; indicates the "no Participant Removal" policy

admin-policy = single-administrator | any-administrator

single-administrator = "single-admin" ; indicates the "Single Administrator"
                                      ; policy

any-administrator = "any-admin" ; indicates the "Any Administrator" policy.
```

The CPM Client SHALL ignore unknown keyword values.

In that case or when the CPM Controlling Function has not indicated a removal-policy in the <keywords> element, the CPM Client shall assume that the “no Participant Removal” policy (i.e. rem-nobody) applies.

Q.1.4 CPM User Roles

The CPM Controlling Function SHALL be able to assign roles to Participant CPM Users of a CPM Group session and a CPM Long-lived Group Session. The following role is defined:

A. Administrator
The Administrator role entitles a CPM User to remove Participants from A CPM Group Session or a CPM Long-lived Group Session.

The CPM User role is assigned via the <roles> element of the <user> element of the conference event package as defined in [RFC4575]. The value of the <roles> element of the conference event package of the CPM Controlling Function SHALL conform with the following ABNF syntax:

```
roles-value = role *("," role)

role = well-known-role / token

well-known-role = Administrator

Administrator = "Administrator"
```

The CPM Client SHALL ignore unknown roles values <roles> element.

The initial assignment of policy in a Group Session is a based on service provider policies.

Note: Extension for the CPM Group Data Management defined in section 6.8 *"CPM Group Data Management"* SHALL be allowed.

The Administrator role SHALL apply to a Participant CPM User, if:

a) the <user> element containing the <yourown> attribute as defined in Appendix M of [OMA-SIMPLE-IM] with the value "true", contains a <roles> element set to a value of "Administrator".