CPM Message Storage

Candidate Version 1.0 – 12 Oct 2010

Open Mobile Alliance

OMA-TS-CPM_MessageStorage-V1_0-20101012-C

© 2010 Open Mobile Alliance Ltd. All Rights Reserved.

Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document. [OMA-Template-Spec-20100101-I]

Contents

Tables

1. Scope

This document provides the technical specifications for the message storage functionality of the CPM Enabler. The document covers the storage of Media Objects, CPM Messages, CPM File Transfer Histories, CPM Session Histories and CPM Conversation Histories in the network and the interactions between client and server components to access the network storage. 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-MSG interface that has been identified in [OMA-CPM-AD]. Also, these technical specifications formally define the expected behaviour of the Message Storage Client and Message Storage Server functional components that have been identified in [OMA-CPM-AD].

2. References

2.1 Normative References

[OMA-CPM-AD]	"Converged IP Messaging Architecture", Open Mobile Alliance™, OMA-AD-CPM-V1_0, URL:http://www.openmobilealliance.org/
[OMA-CPM-RD]	"Converged IP Messaging Requirements", Open Mobile Alliance™, OMA-RD-CPM-V1_0, URL:http://www.openmobilealliance.org/
[OMA-CPM-SD]	"Converged IP Messaging System Description", Open Mobile Alliance™, OMA-TS-CPM_System_Description-V1_0, 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-SEC-CF]	"Security Common Functions Requirements", Open Mobile Alliance, OMA-RD-SEC_CF-V1.0, URL:http//www.openmobilealliance.org/
[RFC2119]	"Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997, URL:http://www.ietf.org/rfc/rfc2119.txt
[RFC2177]	"IMAP IDLE command", B. Leiba, June 1997, URL: http://www.ietf.org/rfc/rfc2177.txt
[RFC2246]	"The TLS Protocol Version 1.0", T. Dierks et al, January 1999, URL:http://www.ietf.org/rfc/rfc2246.txt
[RFC2387]	"The MIME Multipart/Related Content-type", E. Levinson, August 1998, URL:http://www.ietf.org/rfc/rfc2387.txt
[RFC2392]	"Content-ID and Message-ID Uniform Resource Locator", E. Levinson, August 1998, URL: http://www.ietf.org/rfc/rfc2392.txt
[RFC3501]	"Internet Message Access Protocol - version 4rev1", M. Crispin, March 2003, URL:http://www.ietf.org/rfc/rfc3501.txt
[RFC3516]	"IMAP4 Binary Content Extension", L. Nerenberg, April 2003, URL:http://www.ietf.org/rfc/rfc3516.txt
[RFC3862]	"Common Presence and Instant Messaging (CPIM) : Message Format", G. Klyne et al, August 2004, URL: http://www.ietf.org/rfc/rfc3862.txt
[RFC4314]	"IMAP4 Access Control List (ACL) Extension", A. Melnikov, December 2005, URL:http://www.ietf.org/rfc/rfc4314.txt
[RFC4467]	"Internet Message Access Protocol (IMAP) – URLAUTH Extension", M. Crispin, May 2006, URL:http://www.ietf.org/rfc/rfc4467.txt
[RFC4551]	"IMAP Extension for Conditional STORE Operation or Quick Flag Changes Resynchronization", A. Melnikov et al, June 2006, URL:http://www.ietf.org/rfc/rfc4551.txt
[RFC5092]	"IMAP URL Scheme", A. Melnikov, Ed. et al, November 2007, URL:http://www.ietf.org/rfc/rfc5092.txt
[RFC5161]	"The IMAP ENABLE Extension", A. Gulbrandsen, Ed. et al, March 2008, URL:http://www.ietf.org/rfc/rfc5161.txt
[RFC5162]	"IMAP4 extension for Quick Mailbox Resynchronization", A. Melnikov, March 2008, URL:http://www.ietf.org/rfc/rfc5162.txt
[RFC5257]	"Internet Message Access Protocol - ANNOTATE Extension", C. Daboo et al, June 2008, URL:http://www.ietf.org/rfc/rfc5257.txt
[RFC5259]	"Internet Message Access Protocol -CONVERT extension", A. Melnikov, Ed. et al, July 2008, URL:http://www.ietf.org/rfc/rfc5259.txt
[RFC5322]	"Internet Message Format", P. Resnick, October 2008. URL: http://www.ietf.org/rfc/rfc5322.txt
[RFC5423]	"Internet Message Store Event", R. Gellens, et al, March 2009, URL:http://www.ietf.org/rfc/rfc5423.txt
[RFC5464]	"The IMAP METADATA extension", C. Daboo, February 2009, URL:http://www.ietf.org/rfc/rfc5464.txt
[RFC5465]	'The IMAP NOTIFY Extension", A. Gulbrandsen, et al, February 2009, URL:http://www.ietf.org/rfc/rfc5465.txt

[RFC5551] ”Lemonade Notifications Architecture’, R. Gellens, August 2009, URL:http://www.ietf.org/rfc/rfc5551.txt

[RFC5788] “IMAP4 Keyword Registry”, A. Melnikov et al, March 2010, URL:http://www.ietf.org/rfc/rfc5788.txt

2.2 Informative References

[OMADICT] “Dictionary for OMA Specifications”, Open Mobile Alliance™,
OMA-ORG-Dictionary-V2_7, 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

CPM Address	See [OMA-CPM-RD].
CPM Chat Message	See [OMA-CPM-RD].
CPM Conversation	See [OMA-CPM-RD].
CPM Conversation History	See [OMA-CPM-RD].
CPM Conversation Identity	See [OMA-CPM-SD].
CPM File Transfer	See [OMA-CPM-RD].
CPM File Transfer History	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 History	See [OMA-CPM-RD].
CPM Standalone Message	See [OMA-CPM-RD].
CPM User	See [OMA-CPM-RD].
Media Object	See [OMA-CPM-AD].
CPM Session	See [OMA-CPM-RD].
CPM Session Invitation	See [OMA-CPM-RD].
Message Storage Client	See [OMA-CPM-AD].
Message Storage Server	See [OMA-CPM-AD].
Principal	See [OMADICT].
Unique IDentifier	A 32-bit value assigned to each stored object and used with the UID validity value to form a 64-bit value that is unique to a stored object and cannot be referred to any other stored object in the folder or any subsequent folder of the message storage [RFC3501].
UIDVALIDITY	A 32-bit representation of the creation date and time of a folder

3.3 Abbreviations

ACL	Access Control List
CPIM	Common Presence and Instant Messaging
CPM	Converged IP Messaging

IMAP	Internet Message Access Protocol
OMA	Open Mobile Alliance
MIME	Multipurpose Internet Mail Extensions
PSK	Pre Shared Key
SASL	Simple Authentication and Security Layer
TLS	Transport Layer Security
UID	Unique (Message) Identifier
URL	Uniform Resource Locator

4. Introduction

The CPM message storage functionality allows the storage of CPM Messages, CPM File Transfer Histories, CPM Session Histories, CPM Conversation Histories, and any potential Media objects either stand-alone or attached to CPM Messages and CPM Session Histories in a network-based storage on behalf of CPM Users.

The CPM message storage functionality authenticates and authorizes CPM Users to being able to retrieve, organize, set permissions, receive event notifications, synchronize with CPM Users device's local message storage and manage (e.g., copy, remove, move etc.,) the storage objects that are stored on it. It also allows CPM Users to search the storage objects with key words.

4.1   Version 1.0

The CPM 1.0 offers:

- Storage objects

    o CPM Standalone Messages, CPM File Transfer Histories, CPM Session Histories and CPM Conversation Histories

    o any potential Media objects either stand-alone or attached to CPM Standalone Messages, CPM File Transfer Histories and CPM Session Histories

- User authentication and authorization mechanisms

- Operations

    o folder operations e.g., create, list, set active folder, move, remove, search

    o stored object operations e.g., store, fetch, copy, remove, preview

    o metadata operations on stored objects e.g., update metadata

    o access rights on stored objects e.g., set, get, delete

- generate references to stand-alone or attached to CPM Standalone Messages and CPM Session History objects

- allows fetch of stored Media objects by reference

- synchronize between network-based storage and device's local storage

- Notifications

    o notifications about changes in stored resources

5. Common Procedures

5.1 Authorization and Authentication

The IMAPv4 [RFC3501] protocol is used to access the Message Storage Server. This section defines authentication and authorization mechanisms of IMAPv4 used by the Message Storage Server.

5.1.1 Authentication

CPM's message storage functionality supports the two authentication mechanisms listed below, as defined for IMAPv4 [RFC3501];

1.   SASL authentication via the AUTHENTICATE command ([RFC3501]); and

2.   Username/password in plain text authentication via the LOGIN command ([RFC3501]).

The username/password for the second authentication mechanism is separately managed by the CPM service. The password may be pre-configured by the CPM system when the CPM user subscribes to the CPM service.

In addition to these authentication mechanisms, TLS/PSK-TLS, as defined in [RFC2246] and [OMA-SEC-CF], is optional and complementary to simple authentication-only SASL mechanisms or deployed clear-text password login commands. In this way, IMAPv4 can be immune to eavesdropping and hijacking attacks. Using TLS/PSK-TLS, the Message Storage Client also can authenticate the Message Storage Server by checking the certificate supplied by the Message Storage Server.

5.1.2 Authorization

The Message Storage Server uses standard IMAP4 [RFC3501] functionality to enforce access to the stored resources, extended with the possibility for CPM Users to define access control lists (ACL) for their own stored resources.

The ACL management operations and related standard rights are defined in IMAPv4 ACL extension [RFC4314].

In addition to that, the Message Storage Server also allows the use of the IMAP4 URLAUTH extension, as defined in [RFC4467] and [RFC5092]. This URLAUTH extension provides a means by which a Message Storage Client can use URLs carrying authorization information to access limited data on the Message Storage Server.

5.2 Storage Objects

Per the description provided in the CPM System Description [OMA-CPM-SD], there can be five different categories of "storage objects" present in a CPM Message Storage Server. They are folder, message, session history, conversation history and stand-alone media. In this section, these objects are specified in terms of their names and identities, which can be used for various CPM message storage operations in accordance with IMAP4 [RFC3501] and its extensions.

5.2.1 Folder Object

The folder object is realized by the mailbox concept of IMAP4, described in [RFC3501]. The folder object is identified by the name given to it. The CPM folder object aligns with the rules and procedures for names of the mailbox concept of IMAP4, as described in [RFC3501].

5.2.2 Message Object

A message object, matching the message concept described in [RFC3501], is a Message Storage Server stored item. For execution of various IMAP commands, the message object is identified by a 3-component identifier consisting of a folder name, a message identification number associated to the message object and a folder validity value as specified below:

1.   The folder name is the name of the folder in the Message Storage Server where the message is stored,

2. The message identification number is either a message sequence number or a 32-bit Unique Identifier (UID), which is specified according to [RFC3501],

3. The folder validity value or the Unique IDentifier validity (UIDVALIDITY) is another 32-bit value as defined in [RFC3501] distinguishing folder objects of the same name from each other.

5.2.3   Session History Object

The definition and identity specification of the message object SHALL be applicable to the session history object, which consists of a UID, UIDVALIDITY and their values. The session history object is a message formatted according to [] and the clarifications given in below.

The content inserted in the headers of this message is retrieved from the CPM Session Invitation.

The body of this message SHALL be a "multipart/related" body, structured as follows:

- the first body part SHALL contain metadata about the CPM Session History and is formatted according to section 5.2.3.1 "Application/X-CPM-Session Content Type Definition"

- the other body parts SHALL contain the CPM Chat Messages and media that were sent and received via MRSP and RTP during the CPM Session.

An example of a session history object is shown in Appendix D.

Internet Message Format [RFC5322] header	Internet Message Format Parameter status	Content
From	Mandatory	Set to the address of the initiator of the CPM Session, retrieved from the authenticated originator's CPM Address in the SIP INVITE request.
Date	Mandatory	Set to, in order of preference: - the Date of the SIP INVITE request if available, otherwise, - the date and time when the CPM Session recording was started.
Subject	Optional	Set only if "Subject" header is set in the SIP INVITE request.
Conversation-ID	Mandatory	Set to the Conversation-ID of the SIP INVITE request. NOTE: this is a new field, extending the Internet Message Format in accordance with 3.6.8 of [].
Contribution-ID	Mandatory	Set to the Contribution -ID of the SIP INVITE request. NOTE: this is a new field, extending the Internet Message Format in accordance with 3.6.8 of [].

InReplyTo-Contribution-ID	Optional	Set to the InReplyTo -Contribution -ID of the SIP INVITE request. NOTE: this is a new field, extending the Internet Message Format in accordance with 3.6.8 of [].
Content type	Mandatory	Set to the following value : multipart/related;boundary=cpm; type="Application/X-CPM-Session" NOTE: the new "Application/X-CPM-Session" content-type is defined in section 5.2.3.1.
Message body	Mandatory	Contains a list of MIME entities separated by the "cpm" boundary. The first MIME entity is the "root", as defined in [RFC2387] and its content-type is "Application/X-CPM-Session". The other MIME entities are: - the CPM Chat Messages, each formatted as a CPIM MIME entity in accordance with [RFC3862], and, - the Media Objects that were exchanged during the CPM Session.

Table 1: CPM session history object format

5.2.3.1   Application/X-CPM-Session Content Type Definition

This new MIME type is used to store metadata about the CPM Session.

The body of a message with the content-type header set to Application/X-CPM-Session SHALL be formatted as follows:

<session-type>: this element indicates the type of session. It may be set to "Ad-Hoc", "Pre-Defined" or "1-1".

<invited-participants>: this element contains the list of addresses of the invited participants, separated by a semi-column, or the address of the Pre-defined Group.

Next, for each Media Object, metadata information MAY be stored within a <media-object> element as follows:

<media-object>

    <cid>

        cid: set to the Content-ID, as defined in [RFC2392], of the corresponding stored Media Object.

    </cid>

    <sdp>SDP parameters associated with the corresponding stored Media Object.**</sdp>**

</media-object>

The format of the body MAY be extended to store additional metadata information (e.g., participants who joined or left the session).

5.2.4 File Transfer History Object

The definition and identity specification of the message object SHALL be applicable to the file transfer history object, which consists of a UID, UIDVALIDITY and their values.

The file transfer history object is a message formatted according to [] and the clarifications given in below.

The content inserted in the headers of this message is retrieved from the CPM File Transfer invitation.

The body of this message SHALL be a "multipart/related" body, structured as follows:

- the first body part SHALL contain metadata about the CPM File Transfer History and is formatted according to section 5.2.4.1 "Application/X-CPM-File-Transfer Content Type Definition"

- the other body parts SHALL contain the files that were sent and received via MRSP during the CPM File Transfer.

An example of a file transfer history object is shown in Appendix E.

Internet Message Format [RFC5322] header	Internet Message Format Parameter status	Content
From	Mandatory	Set to the address of the initiator of the CPM File Transfer, retrieved from the authenticated originator's CPM Address in the SIP INVITE request.
Date	Mandatory	Set to, in order of preference: - the Date of the SIP INVITE request if available, otherwise, - the date and time when the CPM File Transfer recording was started.
Subject	Optional	Set only if "Subject" header is set in the SIP INVITE request.
Conversation-ID	Mandatory	Set to the Conversation-ID of the SIP INVITE request. NOTE: this is a new field, extending the Internet Message Format, as defined in 3.6.8 of [].
Contribution-ID	Mandatory	Set to the Contribution -ID of the SIP INVITE request. NOTE: this is a new field, extending the Internet Message Format, as defined in 3.6.8 of [].
InReplyTo-Contribution-ID	Optional	Set to the InReplyTo -Contribution -ID of the SIP INVITE request. NOTE: this is a new field, extending the

		Internet Message Format, as defined in 3.6.8 of [].
Content type	Mandatory	Set to the following value : multipart/related;boundary=cpm; type="Application/X-CPM-File-Transfer" NOTE: the new Application/X-CPM File-Transfer content-type is defined in section 5.2.4.1.
Message body	Mandatory	Contains a list of MIME entities separated by the "cpm" boundary. The first MIME entity is the "root", as defined in [RFC2387] and its content-type is "Application/X-CPM-File-Transfer". The other MIME entities are the files that were exchanged during the CPM File Transfer.

Table 2: CPM file transfer history object format

5.2.4.1 Application/X-CPM-File-Transfer Content Type Definition

This new MIME type is used to store metadata about the CPM File Transfer.

The body of a message with the content-type header set to Application/X-CPM-File-Transfer SHALL be formatted as follows :

<file-transfer-type>: this element indicates the type of file transfer. It may be set to "Ad-Hoc", "Pre-Defined" or "1-1".

<invited-participants>: this element contains the list of addresses of the invited participants, separated by a semi-column, or the address of the Pre-defined Group.

Next, for each file, metadata information MAY be stored within a <file-object> element as follows:

<file-object>

   <cid>cid: set to the Content-ID, as defined in [RFC2392], of the corresponding stored file.**</cid>**

   <sdp>SDP parameters associated with the corresponding stored file (e.g., file name).**</sdp>**

</file-object>

5.2.5 Conversation History Object

The conversation history object is viewed as a special kind of sub-folder and is realized via the mailbox concept of IMAP4. A conversation history object stores all items related to a single CPM Conversation as elements (i.e. message objects, session history objects and standalone Media Objects) within the conversation history object. The name of the conversation history object is set to the CPM Conversation Identity associated with the CPM Conversation History that is stored in the conversation history object.

NOTE: The conversation history objects are exposed by the Message Storage Server to the Message Storage Clients; however, the Message Storage Client only shows the contents of the CPM Conversation History (represented as message objects and session history objects) in a manner selected by the CPM User (e.g. in a threaded view).

5.2.6   Stand-alone Media Object

The standalone Media Object is realized by the message concept of IMAP4, whereby the Media Object is wrapped into a MIME formatted object in order to fit into an IMAP4 message. Other than the formatting of the contents, the standalone Media Object is the same as the message object defined in section **Error! Reference source not found.** "*Error! Reference source not found.*", including the definition and naming specification of the message object in terms of UID, next UID and their values.

5.3   Identification of Storage Objects

The combination of a folder name, its stored object's UID, and the folder's UIDVALIDITY MUST permanently and persistently refer to one and only one stored object in a Message Storage Server. In particular, the internal date, size, envelope, body structure, and message texts (HEADER, TEXT and all BODY fetch data items) MUST never change according to [RFC3501]. This requirement does not include message numbers, nor does it include object attributes that can be set by a STORE command (e.g., FLAGS).

Associated with any folder object and conversation history object, there is a next UID value, which is the predicted value that will be assigned to a new storage object in that folder or conversation history. Provided by the CPM Message Storage Server, this next UID is intended to provide a means for the CPM Message Storage Client to determine whether or not any messages have been delivered to the folder since the previous time it checked this value.

> NOTE: The "next UID" value of a folder or conversation history changes whenever a new object is stored in that folder or conversation history.

5.4   Notifications

The Message Storage Server SHALL notify the Message Storage Client of the changes in the stored resources either solicited as a result of a client request or unsolicited and unilaterally according to the IMAP extension for the general notification model [RFC5551].

The data content of a notification may include a wide range of information as follows:

1. Folder size and stored objects status updates, e.g., addition and/or removal of messages to prevent synchronization errors,

2. "Next UID" for the changes to the storage requested by a third entity, e.g., CPM PF,

3. Stored objects' attribute flags, e.g., "\Recent" flag for recently arrived objects in a folder.

5.5   Metadata Structure

CPM's message storage functionality supports a metadata model that consists of two distinct parts:

1. A set of metadata flags that are associated with message objects, file transfer history objects, session history objects and standalone Media Objects. These flags indicate additional state information about the stored object, and

2. A set of metadata annotations that can be associated with all the objects stored in the Message Storage Server. These annotations provide user-defined information that the user associates with these stored objects.

The Message Storage Client and the Message Storage Server SHALL at least support the metadata flags and MAY support the metadata annotations.

With respect to the metadata flags, the Message Storage Client and Message Storage Server SHALL support at least the following flags defined in [RFC3501], [RFC5788] and Appendix C:

• \Seen (message has been read),

• \Answered (message has been answered),

- \Flagged (message is "flagged" for urgent and/or special attention),

- \Deleted (message is "deleted" for removal by later EXPUNGE),

- \Draft (message has not completed composition (marked as a draft),

- \Recent (message is "recently" arrived in this mailbox),

- $MDNSent (A disposition notification has been sent for this message),

- $Forwarded (message has been forwarded), and

- \read-report-sent (A read receipt has been sent for this message) as defined in Appendix C.

NOTE: All flag assignments and operations SHALL be handled according to the procedures specified in [RFC3501].

With respect to the metadata annotations, if supported, the Message Storage Client and the Message Storage Server SHALL support the structure defined in [RFC5464] for metadata annotations for conversation history objects and for folder objects, and the Message Storage Client and the Message Storage Server SHALL support the structure defined in [RFC5257] for message objects, for file transfer history objects, for session history objects and for standalone Media Objects.

6. Procedures at Message Storage Client

The Message Storage Client is a functional component of the CPM enabler, which allows the CPM User to view and manage (store, fetch, delete etc) the resources stored in the Message Storage Server. In addition to that the Message Storage Client notifies the CPM User of any changes to the stored resources in the Message Storage Server (e.g. new message arrived, message got read on another client).

The Message Storage Client SHALL act as anIMAP4 client as defined in [RFC3501]. In addition to that, the Message Storage Client SHALL support the "ACL" IMAP4 extension as defined in [RFC4314], the "URLAUTH" IMAP4 extension as defined in [RFC4467], the "CONDSTORE" IMAP4 extension as defined in [RFC4551], the "ENABLE" IMAP4 extension as defined in [RFC5161RFC5161] and the "QRESYNC" IMAP4 extension as defined in [RFC5162]. Also, the Message Storage Client MAY support the "ANNOTATE" IMAP4 extension as defined in [RFC5257], the "CONVERT" IMAP4 extension as defined in [RFC5259] and/or the "METADATA" IMAP4 extension as defined in [RFC5464].

6.1 General Operations

6.1.1 Authenticate Operation

When the CPM Client needs to authenticate with the Message Storage Server, it SHALL either use the SASL method or the plain-text username/password method.

When the Message Storage Client wants to use the SASL method, the Message Storage Client SHALL send to the Message Storage Server an AUTHENTICATE request as defined in [RFC3501] with the authentication mechanism that it wants to use and then SHALL complete the authentication process as defined in [RFC3501].

When the Message Storage Client wants to use the plain-test username/password method, the Message Storage Client SHALL send to the Message Storage Server a LOGIN request as defined in [RFC3501] with the username and password associated with the Message Storage Client's CPM User.

The CPM Client MAY set up a TLS session prior to authenticating with the Message Storage Server. In order to do so, the Message Storage Client SHALL send to the Message Storage Server a STARTTLS request as defined in [RFC3501] and complete the TLS session setup. The Message Storage Server SHALL support either TLS or PSK-TLS as defined in [RFC2246] and [OMA-SEC-CF] for the TLS protocol negotiations.

6.1.2 Set Active Folder Operation

When a Message Storage Client needs to set a particular folder as the active folder, the Message Storage Client SHALL send to the Message Storage Server a SELECT request as defined in [RFC3501] with the folder name of the folder that is to be set as the active folder.

6.2 Access Control List Operations

6.2.1 Set Access Control List

When a Message Storage Client needs to set the access rights for another Principal on one of the folders of its associated CPM User, the Message Storage Client SHALL send to the Message Storage Server a SETACL request as defined in [RFC4314] with the folder name, the granted access rights and the identifier of the Principal to which access is given.

6.2.2 Get Access Control List

When a Message Storage Client needs to get the access control list on one of the folders of its associated CPM User, the Message Storage Client SHALL send to the Message Storage Server a GETACL request as defined in [RFC4314] with the folder name.

6.2.3   Delete Access Control List

When a Message Storage Client needs to delete the access rights for another Principal on one of the folders of its associated CPM User, the Message Storage Client SHALL send to the Message Storage Server a DELETEACL request as defined in [RFC4314] with the folder name and the identifier of the Principal to which access is given.

6.2.4   Access Rights Retrieval Operations

When a Message Storage Client needs to retrieve the access rights for another Principal on one of the folders of its associated CPM User, the Message Storage Client SHALL send to the Message Storage Server a LISTRIGHTS request as defined in [RFC4314] with a folder name and the identifier of the Principal whose access rights are to be retrieved.

6.3   Message and History Operations

6.3.1   Object Store Operation

When a Message Storage Client needs to store a message object, a file transfer history object, a session history object or a standalone Media Object into a folder on the Message Storage Server, the Message Storage Client SHALL send to the Message Storage Server an APPEND request as defined in [RFC3501] including the name of the folder and the data of the message object, file transfer history object, session history object, or standalone Media Object data. The Message Storage Client MAY include an initial set of metadata flags in the APPEND request towards the Message Storage Server, as defined in [RFC3501]. The Message Storage Client also MAY include a set of metadata annotations in the APPEND request towards the Message Storage Server, as defined in [RFC5257].

> NOTE: The set of metadata flags and metadata annotations associated with the stored object can be changed later using the metadata update operation defined in section **Error! Reference source not found.** "***Error! Reference source not found.***".

6.3.2   Object Fetch Operation

When a Message Storage Client needs to fetch a message object, file transfer history object, session history object or standalone Media Object from the active folder on the Message Storage Server, the Message Storage Client SHALL send to the Message Storage Server a FETCH or a UID FETCH request as defined in [RFC3501] with the UID(s) pointing to a stored object(s) in Message Storage Server.

> NOTE: The Message Storage Client may specify that only specific parts of a message object, file transfer history object, session history object or standalone Media Object are to be fetched, as defined in [RFC3501].

6.3.3   Object Preview Fetch Operation

When a Message Storage Client needs to fetch a preview of a message object, file transfer history object, session history object or standalone Media Object from the active folder on the Message Storage Server, the Message Storage Client SHALL send to the Message Storage Server a CONVERT or a UID CONVERT request as defined in [RFC5259] with the UID pointing to the stored object on the Message Storage Server to be previewed.

> NOTE 1:   The Message Storage Client may specify that only a preview of specific parts of a message object, file transfer history object, session history object or standalone Media Object is to be fetched, as defined in [RFC3501] and [RFC5259].

> NOTE 2:   The CONVERT and UID CONVERT commands can be used to transcode the media type of a MIME part into another media type and/or into the same media type with different encoding parameters.

> NOTE 3:   A "Preview Fetch" operation MAY involve a server-side content adaptation in response to the Message Storage Client's request for the stored object in a compacted or digested form rather than in its original full size and shape.

NOTE 4: Conversions only affect what is sent to the Message Storage Client; the original data in the Message Storage Server SHALL NOT be converted.

6.3.4   Object Copy Operation

When a Message Storage Client needs to copy an existing message object, an existing file transfer history object, an existing session history object or an existing standalone Media Object in the Message Storage Server, the Message Storage Client SHALL send to the Message Storage Server a COPY request as defined in [RFC3501] with the UID pointing to an object to be copied from the Message Storage Server and specifying the destination folder.

6.3.5   Object Remove Operation

When a Message Storage Client needs to remove a stored message object, a file transfer history object, a stored session history object or a stored standalone Media Object, the Message Storage Client SHALL send to the Message Storage Server a STORE request as defined in [RFC3501] with the UID pointing to the stored object to update the flag list associated with the object's data and setting the "\Deleted" flag.

After setting the "\Deleted" flag, the Message Storage Client SHALL send to the Message Storage Server an EXPUNGE request as defined in [RFC3501] in order to permanently remove the message(s) that have been identified for removal from the list of objects with the "\Deleted" flag set.

NOTE: The Message Storage Client can use the EXPUNGE request to permanently remove multiple messages, i.e. there may be multiple STORE commands to set the "Deleted" flag before an EXPUNGE command is executed.

6.4   Folder Operations

6.4.1   Folder Create Operation

When a Message Storage Client needs to create a new folder or a new conversation history object, the Message Storage Client SHALL send to the Message Storage Server a CREATE request as defined in [RFC3501] including the name of the folder.

6.4.2   List Folder Operation

When a Message Storage Client needs to list the contents of the currently selected folder, or conversation history object, the Message Storage Client SHALL send to the Message Storage Server a LIST request as defined in [RFC3501].

6.4.3   Folder Move Operation

NOTE: The folder move operation is also used for renaming folders.

When a Message Storage Client needs to rename a folder or a conversation history object or to move a folder or a conversation history object, the Message Storage Client SHALL send to the Message Storage Server a RENAME request as defined in [RFC3501] including the old name and the new name of the folder or conversation history object to be renamed or moved.

6.4.4   Folder Remove Operation

When a Message Storage Client needs to delete a folder or a conversation history object, the Message Storage Client SHALL send to the Message Storage Server a DELETE request as defined in [RFC3501] including the name of the folder or conversation history object that is to be deleted.

6.4.5   Folder Search Operation

When a CPM Client needs to search in the active folder on the Message Storage Server, the Message Storage Client SHALL send to the Message Storage Server SEARCH request as defined in [RFC3501] including one or more search key data.

6.5 Reference Operations

6.5.1 Generate Reference Operation

When a Message Storage Client needs to generate a reference for (part of) a message object, a file transfer history object, a session history object or a stand-alone Media Object, the Message Storage Client SHALL send to the Message Storage Server a GENURLAUTH request as defined in [RFC4467] including an IMAP URL (as per [RFC5092] and [RFC4467]) pointing to the (part of) the object for which a reference needs to be created.

6.5.2 Fetch by Reference Operation

When a Message Storage Client needs to fetch the (part of) a message object, a file transfer history object, a session history object or a stand-alone media object on the basis of the reference, the Message Storage Client SHALL send to the Message Storage Server a URLFETCH request as defined in [RFC4467].

6.6 Message and History Metadata Management Operations

6.6.1 Metadata Operation

The Message Storage Client SHALL support updating the metadata flags defined for IMAP4 message objects in section **Error! Reference source not found.** *"Error! Reference source not found."* and MAY support updating the metadata annotations defined for IMAP4 mailboxes and IMAP4 message objects in section **Error! Reference source not found.** *"Error! Reference source not found."*.

When a Message Storage Client needs to update the meta data flags of a message object, a file transfer history object, a session history object or a standalone Media Object, the Message Storage Client SHALL send to the Message Storage Server a STORE request as defined in [RFC3501] including the UID of the message object, file transfer history object, session history object or standalone Media Object and the changes to the flags (e.g., seen flag, read flag.)

Additionally, the CPM Message Storage Client MAY use the ANNOTATE extensions as defined in [RFC5257] for the STORE request to provide metadata annotations update.

When a CPM Client needs to update the meta annotations of a folder object or a conversation history object, the Message Storage Client SHALL send to the Message Storage Server a SETMETADATA request as defined in [RFC5464] with the name of the folder and the updated metadata annotations.

6.7 Synchronization

[OMA-CPM-SD] gives a description of the synchronization process between the Message Storage Client and the Message Storage Server. This process only consists of operations described above and therefore needs no further explanation in this section.

While executing these operations, the Message Storage Client SHALL support and use the IMAP4 extensions described in [RFC4551], [RFC5161] and [RFC5162] to get an optimized and quick synchronization between the – potentially offline – Message Storage Client and the Message Storage Server.

6.8 Notification Operations

The IMAP Extension [RFC5465] allows the Message Storage Client to request for solicited notifications about events in specified folders of a Message Storage Server. For making this request, the Message Storage Client SHALL send a NOTIFY command to the Message Storage Server to limit its unsolicited notifications to certain selected folders and certain events such as objects being added to or deleted from those selected folders. Without this Message Storage Client NOTIFY command, an IMAP server will only send information about the changes in the Message Storage Server to the client in the following cases:

1. as the result of a Message Storage Client command such as FETCH responses to a FETCH or STORE command),

2. as unsolicited responses sent just before the end of a command (e.g., EXISTS or EXPUNGE) as the result of changes in other sessions, and

3. during the Message Storage Client's IDLE command

NOTE: Per [RFC2177], the IDLE command provides a way for the client to go into a mode where the Message Storage Server pushes its notifications about the server events in selected folders.

Upon registration and receiving of either solicited or unsolicited notifications from the Message Storage Server, the Message Storage Client SHALL update its corresponding information in the client(s)' locally stored resources.

7. Procedures at Message Storage Server

The Message Storage Server is a functional component of the CPM enabler, which allows authorized and/or authenticated principals (such as Message Storage Clients or CPM Participating Functions) to access a resource in the Message Storage Server.

The Message Storage Server SHALL act as an IMAP4 server as defined in [RFC3501]. In addition to that, the Message Storage Server SHALL support the "ACL" IMAP4 extension as defined in [RFC4314], the "URLAUTH" IMAP4 extension as defined in [RFC4467], the "CONDSTORE" IMAP4 extension as defined in [RFC4551], the "ENABLE" IMAP4 extension as defined in [RFC5161] and the "QRESYNC" IMAP4 extension as defined in [RFC5162]. Also, the Message Storage Server MAY support the "CONVERT" IMAP4 extension as defined in [RFC5259] and/or the "METADATA" IMAP4 extension as defined in [RFC5464].

7.1 General Operations

7.1.1 Authenticate Operation

Upon receiving a STARTTLS request, the Message Storage Server SHALL process the request and return a response according to the STARTTLS command as defined in [RFC3501]. The Message Storage Server SHALL at least support TLS and PSK-TLS as defined in [RFC2246] and [OMA-SEC-CF] for the TLS protocol negotiations.

Upon receiving an AUTHENTICATE request, the Message Storage Server SHALL process the request and return a response according to the AUTHENTICATE command as defined in [RFC3501] and then complete the authentication process as defined in [RFC3501].

Upon receiving a LOGIN request, the Message Storage Server SHALL process the request and return a response according to the LOGIN command as defined in [RFC3501].

7.1.2 Set Active Folder Operation

Upon receiving a SELECT request, the Message Storage Server SHALL process the request and return a response according to the SELECT command as defined in [RFC3501].

7.2 Access Control List Operations

7.2.1 Set Access Control List

Upon receiving a SETACL request, the Message Storage Server SHALL process the request and return a response according to the SETACL command as defined in [RFC4314].

7.2.2 Get Access Control List

Upon receiving a GETACL request, the Message Storage Server SHALL process the request and return a response according to the GETACL command as defined in [RFC4314].

7.2.3 Delete Access Control List

Upon receiving a DELETEACL request, the Message Storage Server SHALL process the request and return a response according to the DELETEACL command as defined in [RFC4314].

7.2.4 Access Rights Retrieval Operations

Upon receiving a LISTRIGHTS request the Message Storage Server SHALL process the request and return response according to the descriptions as defined in [RFC4314].

7.3   Message and History Operations

7.3.1   Object Store Operation

Upon receiving an APPEND request including the object to store, the Message Storage Server SHALL handle the request according to the APPEND command as defined in [RFC3501] and store the specified object to the end of the specified destination folder.

7.3.1.1   Handling Deferred CPM Message Objects

Upon receiving a request for storing a Deferred CPM Message object that is associated with an expiry time, the Message Storage Server SHALL:

1. handle the request according to the APPEND command as defined in [RFC3501] and store the object to the end of the specified destination folder designated for temporarily holding deferred messages and

2. notify the Message Storage Client of the arrival of the new message.

The stored message object will remain in the designated folder until either it is moved by the Message Storage Client's corresponding commands or it reaches its deferred expiry time and is deleted from the folder.

7.3.2   Object Fetch Operation

Upon receiving a FETCH request with the UID pointing to a stored object, the Message Storage Server SHALL handle the request according to the FETCH or UID FETCH command as defined in [RFC3501].

7.3.3   Object Preview Fetch Operation

Upon receiving a CONVERT or UID CONVERT request with the UID pointing to a stored object of the Message Storage Server, the Message Storage Server SHALL handle the preview request for the supplied format and dimensions according to the CONVERT or UID CONVERT command as defined in [RFC5259]

7.3.4   Object Copy Operation

Upon receiving a COPY request with the UID pointing to a stored object of the Message Storage Server, the Message Storage Server SHALL copy the specified Message(s) or Message History to the end of the specified destination folder according to the COPY command as defined in [RFC3501]

7.3.5   Object Remove Operation

Upon receiving an IMAP STORE command with a UID pointing to the stored object to update the flag list of the object's data to include the "\Deleted" flag, the Message Storage Server SHALL handle the flagging request by setting the stored object's "\Deleted" flag according to the STORE command as defined in [RFC3501].

Upon receiving an EXPUNGE request to remove all stored objects from the CPM User's Message Storage Server that have the "\Deleted" flag set for removal, the Message Storage Server SHALL handle the request to permanently remove these stored objects according to the EXPUNGE command as defined in [RFC3501].

7.4   Folder Operations

7.4.1   Folder Create Operation

Upon receiving a CREATE request, the Message Storage Server SHALL create the folder (e.g., mailbox) with the requested name according to the CREATE command as defined in [RFC3501].

7.4.2 List Folders Operation

Upon receiving a LIST request, the Message Storage Server SHALL determine and return the names of all folders of the CPM User on the Message Storage Server according to the LIST command as defined in [RFC3501].

7.4.3 Folder Move Operation

NOTE: The folder move operation is also used for renaming folders.

Upon receiving a RENAME request with the request-folder name and its new name, the Message Storage Server SHALL rename the indicated folder according to the RENAME command as defined in [RFC3501].

7.4.4 Folder Remove Operation

Upon receiving a DELETE request with the request-folder name, the Message Storage Server SHALL remove the indicated folder according to the DELETE command as defined in [RFC3501].

7.4.5 Folder Search Operation

Upon receiving a SEARCH request, with one or more search key data, the Message Storage Server SHALL handle the request according to the IMAP SEARCH command as defined in [RFC3501].

7.5 Reference Operations

7.5.1 Generate Reference Operation

Upon receiving a GENURLAUTH request, the Message Storage Server SHALL process the request and return a response according to the GENURLAUTH command as defined in [RFC4467].

7.5.2 Fetch by Reference Operation

Upon receiving a URLFETCH request, the Message Storage Server SHALL process the request and return a response according to the URLFETCH command as defined in [RFC4467].

7.6 Metadata Update Operation

The Message Storage Server SHALL support the metadata flags defined for IMAP4 message objects in section **Error! Reference source not found.** *"Error! Reference source not found."* and MAY support the metadata annotations defined for IMAP4 mailboxes and IMAP4 message objects in section **Error! Reference source not found.** *"Error! Reference source not found."*.

Upon receiving a STORE request with the UID addressing a stored object of the Message Storage Server, the Message Storage Server SHALL update the metadata flags as indicated in the request according to the STORE command as defined in [RFC3501]. The Message Storage Server MAY handle any metadata annotations specified in the STORE request according to the ANNOTATE IMAP4 extension defined in [RFC5257].

Upon receiving a SETMETADATA request with the UID addressing a stored object of the Message Storage Server, the Message Storage Server SHALL update the metadata annotations for the indicated updates (add, remove & modify) according to the SETMETADATA command as defined in [RFC5464]

7.7 Message and History Synchronization Operations

[OMA-CPM-SD] gives a description of the synchronization process between the Message Storage Client and the Message Storage Server. This process only consists of operations described above and therefore needs no further explanation in this section.

In addition to that, the Message Storage Server SHALL support the IMAP4 extensions described in [RFC4551], [RFC5161] and [RFC5162] to allow the Message Storage Client to have an optimized and quick synchronization process.

7.8 Notifications Operations

Upon receiving the Message Storage Client's NOTIFY command and/or occurrence of changes in the Message Storage Server's resources, the Message Storage Server SHALL reflect the presence of changes via sending Notification messages. The changes in the Message Storage Server's resources MAY include one or more of the following specific operations per [RFC5423]:

1. message addition and deletion

2. Message flags, e.g., read, clear

3. Access accounting, e.g., login, logout

4. Folder management, e.g., create, delete, rename

The notification messages SHALL be sent by the Message Storage Server to the Message Storage Client upon the occurrence of the following events:

1. Changes in the stored objects due to performing any operations by the Message Storage Server on the stored objects during the period when the CPM Message Storage Client is registered and the notification feature is activated.

2. Changes occurred since the CPM Message Storage Client's last de-registration or CPM User's deactivation of the notification feature.

NOTE: While IMAP IDLE, NOTIFY, or other status change indications serve the purpose of notification, they are not necessarily for notifications only. They are part of the greater synchronization mechanism.

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 <current version> History

Document Identifier	Date	Sections	Description
Draft Versions OMA-TS-CPM_MessageStorage-V1_0	17 Nov 2008	All	Initial Draft
	03 Mar 2009	5, 6	1. Incorporates the following CRs: OMA-MWG-CPM-2009-0064R02-CR_TS_Message_Storage_Detailed_TOC 2. All new Chapters moved one level up in the tree structure
	15 Sep 2009	6	Incorporates following CRs: 1. OMA-MWG-CPM-2009-0402R01-CR_TS_MSG_STOR_Srvr_Procedures 2. OMA-MWG-CPM-2009-0419R03-CR_TS_MSG_STOR_Client_Procedures
	20 Nov 2009	2.1, 5.1.1, 5.4, 6.1.1.3, 6.1.1.5, 6.1.3.1, Appendix B	Incorporates following CRs: 1. OMA-MWG-CPM-2009-0498R03-CR_TS_MSG_STOR_SCR 2. OMA-MWG-CPM-2009-0500R02-CR_TS_MSG_STOR_Preview_Fetch_Ope 3. OMA-MWG-CPM-2009-0511R01-CR_TS_MSG_STOR_Object_Remote_Operation 4. OMA-MWG-CPM-2009-0516-CR_TS_MSG_STOR_Metadata_Operation 5. OMA-MWG-CPM-2009-0551R01-CR_TS_MSS_ACL_Management 6. OMA-MWG-CPM-2009-0553R01-CR_TS_MSS_Rights_Retrieval 7. OMA-MWG-CPM-2009-0602-CR_TS_MSS_Metadata_Structure
	22 Dec 2009	All	Incorporated following CRs: 1. OMA-MWG-CPM-2009-0408R03-CR_CPM_TS_MessageStorage_Security 2. OMA-MWG-CPM-2009-0548R03-CR_TS_STORE_Command_for_Metadata_Update 3. OMA-MWG-CPM-2009-0601R02-CR_TS_MSS_Error_Handling_Procedure 4. OMA-MWG-CPM-2009-0644R01-CR_TS_MSG_STOR_Notification_Procedure 5. OMA-MWG-CPM-2009-0645R02-CR_TS_MSS_Storage_Objects 6. OMA-MWG-CPM-2009-0676-CR_MessageStorage_TS__Reference_section 7. OMA-MWG-CPM-2009-0678-CR_TS_MessageStorage_Definitions_Section 8. OMA-MWG-CPM-2009-0683-CR_TS_message_storage_Abbreviation_section
	03 Mar 2010	All	Incorporated as per agreement in: OMA-MWG-CPM-2010-0105R02-INP_CONRR_TS_Message_Storage_Editorial_Comments

Document Identifier	Date	Sections	Description
	04 Mar 2010	2.1, all sections under 5.1, 6.2.3,	Incorporated following CRs: 1. OMA-MWG-CPM-2010-0052R01-CR_TS_Message_Storage_Section_2.1 2. OMA-MWG-CPM-2010-0053R01-CR_TS_Message_Storage_Section_6.2.3 3. OMA-MWG-CPM-2010-0090R02-CR_TS_CONR_Comment_F038 4. OMA-MWG-CPM-2010-0129-CR_TS_MS_CONR_C301 5. OMA-MWG-CPM-2010-0140-CR_TS_MSGSTOR_CONRR_F037 6. OMA-MWG-CPM-2010-0141-CR_TS_MSGSTOR_CONRR_F041 7. OMA-MWG-CPM-2010-0146R02-CR_TS_MSGSTOR_CONRR_Authorisation_Section
	12 Mar 2010	All under 6.1.1 and 6.1.2, 5.2.2, 5.3, 6.1.2.4, 6.2.2.4, B.1, B.2, 6.2.5,	Incorporated following CRs: 1. OMA-MWG-CPM-2010-0169R02-CR_TS_MSGSTOR_CONRR_Message_Storage_Client_Comments 2. OMA-MWG-CPM-2010-0185R01-CR_TS_MS_CONR_Comments_F085_F086 3. OMA-MWG-CPM-2010-0187R01-CR_TS_MS_CONR_Comment_F094 4. OMA-MWG-CPM-2010-0210-CR_TS_MSGSTOR_CONRR_F141_F168 5. OMA-MWG-CPM-2010-0211-CR_TS_MSGSTOR_CONRR_F182_F184 6. OMA-MWG-CPM-2010-0212-CR_TS_MSGSTOR_CONRR_F125_F129 7. OMA-MWG-CPM-2010-0213-CR_TS_MSGSTOR_CONRR_F175_F176_F177
	21 Apr 2010	1, 2.1, 3.2, 3.3, 5.5	Incorporated following CRs: 1. OMA-COM-CPM-2010-0209-CR_TS_MSGSTOR_CONRR_F007_F008 2. OMA-COM-CPM-2010-0337-CR_TS_MS_CONR_Comment_F036 (no changes) 3. OMA-COM-CPM-2010-0339-CR_TS_MS_CONR_F102_to_F106 4. OMA-COM-CPM-2010-0369-CR_CONRR_F013_and_F018 5. OMA-COM-CPM-2010-0370R01-CR_CONRR_F023 6. OMA-COM-CPM-2010-0371-CR_CONRR_F027 7. OMA-COM-CPM-2010-0372-CR_CONRR_F111
	14 May 2010	2.1, 5.1, 5.1.2, 5.2.3, 5.4, 6.1.1, 6.1.2, 6.1.6, 6.2.1, 6.2.2.1, 6.2.6	Incorporated following CRs: 1. OMA-MWG-CPM-2010-0186R04-CR_TS_MS_CONR_Comments_F088_F089_F091 2. OMA-COM-CPM-2010-0338R02-CR_TS_MS_CONR_F095_to_F101 3. OMA-COM-CPM-2010-0416R01-CR_TS_MSS_CONR_Comment_F131 4. OMA-COM-CPM-2010-0426-CR_TS_MSG_CONRR_F039_F040_F043 5. OMA-COM-CPM-2010-0427-CR_TS_MSG_Client_Terminology 6. OMA-COM-CPM-2010-0428-CR_TS_MSG_CONRR_F149_F150 7. OMA-COM-CPM-2010-0429-CR_TS_MSG_CONRR_F154_F156

Document Identifier	Date	Sections	Description
	19 May 2010	2.1, 5.2.2.1, 5.2.5, 5.3, 5.5, 6.1, 6.1.2.1, 6.1.2.2, 6.1.2.3, 6.1.4.1, 6.2.4, B.1, B.2	Incorporated following CRs: 1. OMA-COM-CPM-2010-0411R01-CR_TS_MS_CONR_F107_to_F110 2. OMA-COM-CPM-2010-0449R01-CR_TS_MSG_CONRR_F121_F146_F171_F183_F185 3. OMA-COM-CPM-2010-0450-CR_TS_MSG_CONRR_F124_F128 4. OMA-COM-CPM-2010-0451-CR_TS_MSG_CONRR_F120 5. OMA-COM-CPM-2010-0484-CR_TS_MSG_CONRR_F114_F115 6. OMA-COM-CPM-2010-0485-CR_TS_MSG_CONRR_F087 7. OMA-COM-CPM-2010-0487-CR_TS_MSG_CONRR_F093
	20 May 2010		Incorporated following CRs: 1. OMA-COM-CPM-2010-0486R02-CR_TS_MST_CONRR_F092 2. OMA-COM-CPM-2010-0488R01-CR_TS_MSG_CONRR_F083_F084
	31 May 2010	2.1, 5.1.1, 6.1.1, 6.2.1	Incorporated following CRs: 1. OMA-COM-CPM-2010-0516-CR_TS_MSG_CONRR_F071_to_F079 2. OMA-COM-CPM-2010-0517-CR_TS_MSG_CONRR_F070_F113 OMA-COM-CPM-2010-0518-CR_TS_MSG_Editorial_Glitches
	19 Jul 2010	2.1, 6.1, 6.1.6, 6.2, 6.2.6	Incorporated following CRs: 1. OMA-COM-CPM-2010-0646-CR_TS_MSG_CONRR_F147_F148_F174
	10 Aug 2010	4, 4.1, 5.5, 7.3.5	Incorporated following CRs: 1. OMA-COM-CPM-2010-0695-CR_TS_MS_CONR_F002_F003_F178 2. OMA-COM-CPM-2010-0730-CR_TS_MSG_CONRR_F112 3. OMA-COM-CPM-2010-0772R01-CR_MSG_TS_CONRR_D234 4. OMA-COM-CPM-2010-0777R01-CR_MSG_TS_CONRR_F028_to_F033
	27 Aug 2010	All	Incorporated following CRs: 1. OMA-COM-CPM-2010-0813R01-CR_TS_MSG_Editorial_Cleanup 2. OMA-COM-CPM-2010-0815-CR_TS_MSG_Fix_Object_Remove_Procedure 3. OMA-COM-CPM-2010-0816-CR_TS_MSG_Fix_Read_Report_Sent_Flag 4. OMA-COM-CPM-2010-0817-CR_TS_MSG_Fix_SCR_Tables
	07 Sep 2010	All	Fixed all broken cross references Incorporated following CRs: 1. OMA-COM-CPM-2010-0758R05-CR_TS_MSG_Session_History_Format 2. OMA-COM-CPM-2010-0834R01-CR_TS_MSS_Defer_and_Store 3. OMA-COM-CPM-2010-0883-CR_TS_MSG_Companion_to_758R04 4. OMA-COM-CPM-2010-0884-CR_TS_MSG_Add_Reference_Operations
Candidate Version OMA-TS-CPM_MessageStorage-V1_0	12 Oct 2010	All	Status changed to Candidate by TP: OMA-TP-2010-0420-INP_CPM_V1_0_ERP_for_Candidate_Approval

Appendix B.   Static Conformance Requirements   (Normative)

The notation used in this appendix is specified in [**Error! Reference source not found.**].

B.1   SCR for Message Storage Client

Item	Function	Reference	Requirement
CPM-TS-MS-C-001_M	Authentication – setting up TLS connection	6.1.1	
CPM-TS-MS-C-002_M	Authentication – SASL method	6.1.1	
CPM-TS-MS-C-003_M	Authentication – username/password method	6.1.1	
CPM-TS-MS-C-004_M	Requesting to set an active folder	6.1.2	
CPM-TS-MS-C-005_M	Setting an ACL	6.2.1	
CPM-TS-MS-C-006_M	Getting an ACL	6.2.2	
CPM-TS-MS-C-007-M	Deleting an ACL	6.2.3	
CPM-TS-MS-C-008_M	Retrieving access rights	6.2.4	
CPM-TS-MS-C-009_M	Requesting to store an object(i.e. Message and Message histories)	6.3.1	
CPM-TS-MS-C-010_M	Requesting to fetch message(s)	6.3.2	
CPM-TS-MS-C-011_O	Requesting to fetch a preview object	6.3.3	
CPM-TS-MS-C-012_M	Requesting to copy objects to another folder	6.3.4	
CPM-TS-MS-C-013_M	Requesting to remove objects	6.3.5	
CPM-TS-MS-C-014_M	Requesting to create a folder	6.4.1	
CPM-TS-MS-C-015_M	Requesting to get a list of folders	6.4.2	
CPM-TS-MS-C-016_M	Requesting to rename the folder	6.4.3	
CPM-TS-MS-C-017_M	Requesting to remove the folder	6.4.4	
CPM-TS-MS-C-018_M	Requesting to search contents by a few key words	6.4.5	
CPM-TS-MS-C-019_M	Updating metadata flags of a stored object	6.5.1	
CPM-TS-MS-C-020_O	Updating metadata annotations of a stored object (ANNOTATE extension)	6.5.1	
CPM-TS-MS-C-021_O	Updating metadata annotations of a folder	6.6	

Item	Function	Reference	Requirement
	(METADATA extension)		
CPM-TS-MS-C-022_M	Synchronization	6.6	
CPM-TS-MS-C-023_O	Notifications	6.7	

B.2   SCR for Message Storage Server

Item	Function	Reference	Requirement
CPM-TS-MS-S-001_M	Authentication – setting up TLS connection	7.1.1	
CPM-TS-MS-S-002_M	Authentication – SASL method	7.1.1	
CPM-TS-MS-S-003_M	Authentication – username/password method	7.1.1	
CPM-TS-MS-S-004_M	Requesting to set an active folder	7.1.2	
CPM-TS-MS-S-005_M	Setting an ACL	7.2.1	
CPM-TS-MS-S-006_M	Getting an ACL	7.2.2	
CPM-TS-MS-S-007-M	Deleting an ACL	7.2.3	
CPM-TS-MS-S-008_M	Retrieving access rights	7.2.4	
CPM-TS-MS-S-009_M	Storing objects to a folder	7.3.1	
CPM-TS-MS-S-010_M	Sending objects	7.3.2	
CPM-TS-MS-S-011_O	Converting the object as requested	7.3.3	
CPM-TS-MS-S-012_M	Copying the objects to the another folder	7.3.4	
CPM-TS-MS-S-013_M	Removing the object	7.3.5	
CPM-TS-MS-S-014_M	Creating a folder with a requested folder name	7.4.1	
CPM-TS-MS-S-015_M	Sending the name of all folders	7.4.2	
CPM-TS-MS-S-016_M	Renaming the folder	7.4.3	
CPM-TS-MS-S-017_M	Deleting the folder	7.4.4	
CPM-TS-MS-S-018_M	Searching objects by a few keywords	7.4.5	
CPM-TS-MS-S-019_M	Updating metadata flags for a stored object	7.5	
CPM-TS-MS-S-020_O	Updating metadata annotations of a stored object (ANNOTATE extension)	6.2.3	
CPM-TS-MS-S-021_O	Updating metadata annotations of a folder (METADATA extension)	7.5	
CPM-TS-MS-S-022_M	Processing the synchronization	7.6	
CPM-TS-MS-S-023_O	Notifications	7.7	

Appendix C.   Appendix C. CPM-defined IMAP Flag Extensions

C.1   \read-report-sent

\ read-report-sent is a CPM-defined IMAP flag extension associated with a Message Object stored in Message Storage Server.

The definition of the flag is:

```
\read-report-sent
   Read report is sent
```

Formal syntax of \read-report-sent is:

```
flag-extension = "\read-report-sent"
```

 Example:

```
C: A003 STORE 2:4 +FLAGS (\read-report-sent)
```

\ read-report-sent status flag is visible to the Message Storage Client but is masked to the CPM User. This flag is used to record whether a read report has been sent or not for a message object in the Message Storage Server.

Appendix D.   Example of Session History Object

Below is an example of a 1-1 session history object with the following content:

-   1 CPM Chat Message.

-   1 audio stream.

```
From: John Doe <jdoe@machine.example>
Date: Fri, 21 Nov 1997 09:55:06 -0600
Subject: the weather will be fine today
Conversation-ID: f81d4fae-7dec-11d0-a765-00a0c91e6bf6
Contribution-ID: abcdef-1234-5678-90ab-cdef01234567
InReplyTo-Contribution-ID: 01234567-89ab-cdef-0123-456789abcdef
Content-type: multipart/related;boundary=cpm; type="Application/X-CPM-Session"

--cpm
Content-Type: Application/X-CPM-Session

<session-type>1-1</session-type>
<invited-participants>jdoe@machine.example; sip:alice@example.com</invited-
participants>

<media-object>
      <cid>cid:<1234@example.com></cid>
      <sdp>
        a=sendonly
        m=audio 49170 RTP/AVP 0 97
      </sdp>
</media-object>

--cpm
Content-type: Message/CPIM
From: John Doe <jdoe@machine.example>
To: Alice <sip:alice@example.com>
DateTime: 2000-12-13T13:40:00-08:00
Content-type: text/plain; charset=utf-8


Here is the text of my message.

--cpm
Content-Type: audio/basic
Content-Transfer-Encoding: base64
Content-ID:<1234@example.com>

      ... base64-encoded 8000 Hz single-channel
          mu-law-format audio data goes here ...
```

Appendix E.   Example of File Transfer History Object

Below is an example of a 1-1 file transfer history object in which one file was received.

```
From: John Doe <jdoe@machine.example>
Date: Fri, 21 Nov 1997 09:55:06 -0600
Conversation-ID: f81d4fae-7dec-11d0-a765-00a0c91e6bf6
Contribution-ID: abcdef-1234-5678-90ab-cdef01234567
InReplyTo-Contribution-ID: 01234567-89ab-cdef-0123-456789abcdef
Content-type: multipart/related;boundary=cpm; type="Application/X-CPM-File-Transfer"

--cpm
Content-Type: Application/X-CPM-File-Transfer

<file-transfer-type>1-1</file-transfer-type>
<invited-participants>jdoe@machine.example; sip:alice@example.com</invited-
participants>

<file-object>
     <cid>cid:<1234@example.com></cid>
     <sdp>
       i=This is my latest picture
       a=sendonly
       a=file-selector:name:"My picture.jpg" type:image/jpeg size:4092
       a=file-disposition:render
       a=file-date:creation:"Mon, 15 May 2006 15:01:31 +0300"
     </sdp>
</file-object>

--cpm
Content-Type: image/jpeg
Content-Transfer-Encoding: binary
Content-ID: <1234@example.com>

   ... My picture.jpg...
```