Multimedia Messaging Service
Client Transactions

Approved Version 1.1 – 15 Jul 2004

Open Mobile Alliance
OMA-WAP-MMS-CTR-V1_1-20040715-A

Continues the Technical Activities
Originated in the WAP Forum

Contents

1. Scope

The Wireless Application Protocol (WAP) is a result of continuous work to define an industry-wide specification for developing applications that operate over wireless communication networks. The scope for the WAP Forum is to define a set of specifications to be used by service applications. The wireless market is growing very quickly, and reaching new customers and services. To enable operators and manufacturers to meet the challenges in advanced services, differentiation and fast/flexible service creation WAP Forum defines a set of protocols in transport, security, transaction, session and application layers. For additional information on the WAP architecture, please refer to "*Wireless Application Protocol Architecture Specification"* [WAPARCH].

Multimedia Messaging Service (MMS) is a system application by which a WAP client is able to provide a messaging operation with a variety of media types. The service is described in terms of actions taken by the WAP MMS Client and its service partner, the MMS Proxy-Relay, a device that operates as a WAP Origin Server for this specialised service. This specification defines the operational flow of the MMS Protocol Data Units (PDUs) that transit between the MMS Client and the MMS Proxy-Relay. The format of the specific PDUs is described in the "*WAP MMS Encapsulation Protocol*" [MMSENCAPS].

This document is part of the WAP MMS version 1.1 specification suite and complies with the requirements and service behaviours described in the technical specifications of the 3rd Generation Partnership Project (3GPP). These include the service aspects of MMS and the functional description of MMS which are contained in [TS23140].

For information about the MMS Architecture, the reader is advised to become familiar with the "*WAP MMS Architecture Overview*" [MMSARCH].

2. References

2.1 Normative References

[CREQ] "Specification of WAP Conformance Requirements". Open Mobile Alliance™. WAP-221-CREQ-20010425-a. URL: http://www.openmobilealliance.org

[MMSENCAPS] "Wireless Application Protocol, MMS Encapsulation Protocol, Version 1.1", WAP-276-MMSEncapsulation, WAP Forum, tbd-March-2002. URL: http://www.openmobilealliance.org

[PUSHMSG] "Wireless Application Protocol, Push Message Specification", WAP-251-PushMessage, WAP Forum, 25-April-2001. URL: http://www.openmobilealliance.org

[PUSHOTA] "Wireless Application Protocol, Push OTA Protocol", WAP-235-PushOTA, WAP Forum, 25-April-2001. URL: http://www.openmobilealliance.org

[PUSHPAP] "Wireless Application Protocol, Push Access Protocol Specification", WAP-247-PAP, WAP Forum, 29-April-2001. URL: http://www.openmobilealliance.org

[RFC1766] "Tags for the Identification of Languages", H. Alvestrand, March 1995. URL: http://www.ietf.org/rfc/rfc1766.txt

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

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

[RFC2234] "Augmented BNF for Syntax Specifications: ABNF". D. Crocker, Ed., P. Overell. November 1997. URL:http://www.ietf.org/rfc/rfc2234.txt

[RFC2396] "Uniform Resource Identifiers (URI): Generic Syntax", Berners-Lee T., Fielding R., Masinter L., August 1998. URL: http://www.ietf.org/rfc/rfc2396.txt

[RFC2616] "Hypertext Transfer Protocol HTTP/1.1", Fielding R., Gettys J., Mogul J., Frystyk H., Masinter L., Leach P., Berners-Lee T., June 1999. URL: http://www.ietf.org/rfc/rfc2616.txt

[RFC2817] "Upgrading to TLS Within HTTP/1.1", R. Khare et al., May 2000. URL: http://www.ietf.org/rfc/rfc2817.txt

[UAPROF] "Wireless Application Protocol, User Agent Profile", WAP-248-UAProf, WAP Forum, 30-May-2001. URL: http://www.openmobilealliance.org

[WAPWSP] "Wireless Application Protocol, Wireless Session Protocol Specification", WAP-230-WSP, WAP Forum, 5-July-2001. URL: http://www.openmobilealliance.org

[WDP] "Wireless Application Protocol, Wireless Datagram Protocol", WAP-259-WDP, WAP Forum, 14-June-2001. URL: http://www.openmobilealliance.org

[WP-HTTP] "Wireless Application Protocol, Wireless Profiled HTTP", WAP-229-HTTP, WAP Forum, 29-March-2001. URL: http://www.openmobilealliance.org

[WP-TCP] "Wireless Application Protocol, Wireless Profiled TCP", WAP-225-TCP, WAP Forum, 31-March-2001. URL: http://www.openmobilealliance.org

[WP-TLS] "Wireless Application Protocol, TLS Profile and Tunneling", WAP-219-TLS, WAP Forum, 11-April-2001. URL: http://www.openmobilealliance.org

[WTLS] "Wireless Application Protocol, Wireless Transport Layer Security Specification", WAP-261-WTLS, WAP Forum, 6-April-2001. URL: http://www.openmobilealliance.org

[WTP] "Wireless Application Protocol, Wireless Transaction Protocol", WAP-224-WTP, WAP Forum, 10-July-2001. URL: http://www.openmobilealliance.org

2.2 Informative References

[3G2VID] 3GPP2 S.R0021: “Video Streaming Services – Stage 1”

[MMSARCH] “Wireless Application Protocol, MMS Architecture Overview, Version 1.1”, WAP-274-MMSArchOverview, WAP Forum, 5-March-2002. URL: http://www.openmobilealliance.org

[PSSCOD] 3GPP TS 26.234 Rel-4: "Packet-switched Streaming Service (PSS); Protocols and Codecs".

[PSSGEN] 3GPP TS 26.233 Rel-4: "Packet-switched Streaming Service (PSS); General Description".

[PUSHARCH] "Wireless Application Protocol, Push Architecture Overview", WAP-250-PushArchOverview, WAP Forum, 3-July-2001. URL: http://www.openmobilealliance.org

[TS22140] “Multimedia Messaging Service: Service aspects; Stage 1”, 3rd Generation Partnership Project TS 22.140 Release 4. URL: http://www.3gpp.org/ftp/Specs/

[TS23140] “Multimedia Messaging Service: Functional description; Stage 2”, 3rd Generation Partnership Project TS 23.140 Release 4. URL: http://www.3gpp.org/ftp/Specs/

[WAPARCH] “Wireless Application Protocol, Architecture Specification”, WAP-210-WAPArch, WAP Forum, 12-July-2001. 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

This section introduces a terminology that will be used throughout this document.

Forwarding MMS Client

An MMS Client forwarding an MM to another recipient without prior download.

Multimedia Messaging Service (MMS)

A system application by which a WAP client is able to provide a messaging operation with a variety of media types.

MMS Client

The MMS service endpoint located on the WAP client device.

MMS Proxy-Relay

A server which provides access to various messaging systems. If the MMS Proxy-Relay operates as a WAP origin server it may be able to utilise features of the WAP system.

MMS Server

A server that provides storage and operational support for the MMS service.

MMS_M Link

The interface between the MMS Client and its service partner, the MMS Proxy-Relay.

Terminal

A WAP client device.

Transaction

One or more PDU exchanges that collectively are considered logically separate from other PDU exchanges.

WAP Origin Server

A server that can deliver appropriate content upon request from a WAP client.

3.3 Abbreviations

Email	Electronic mail
HTTP	HyperText Transfer Protocol
IANA	Internet Assigned Numbers Authority
ID	Identifier
MIME	Multipurpose Internet Mail Extensions
MM	Multimedia Message
MMS	Multimedia Messaging Service
OTA	Over The Air
PDF	Portable Document Format
PEP	Performance Enhancing Proxy

PDU	Protocol Data Unit
PGP	Push Proxy Gateway
RDF	Resource Description Format
RFC	Request For Comments
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UAProf	User Agent Profile
URI	Uniform Resource Identifier
WAP	Wireless Application Protocol
WSP	Wireless Session Protocol
WTLS	Wireless Transport Layer Security
XML	eXtensible Markup Language

4. Introduction

This section is informative.

The Multimedia Messaging Service (MMS) is intended to provide non-real-time messaging services to consumers utilising WAP technologies. It is an application level service that fits into the current WAP architecture. The following figure shows the general MMS Architecture.

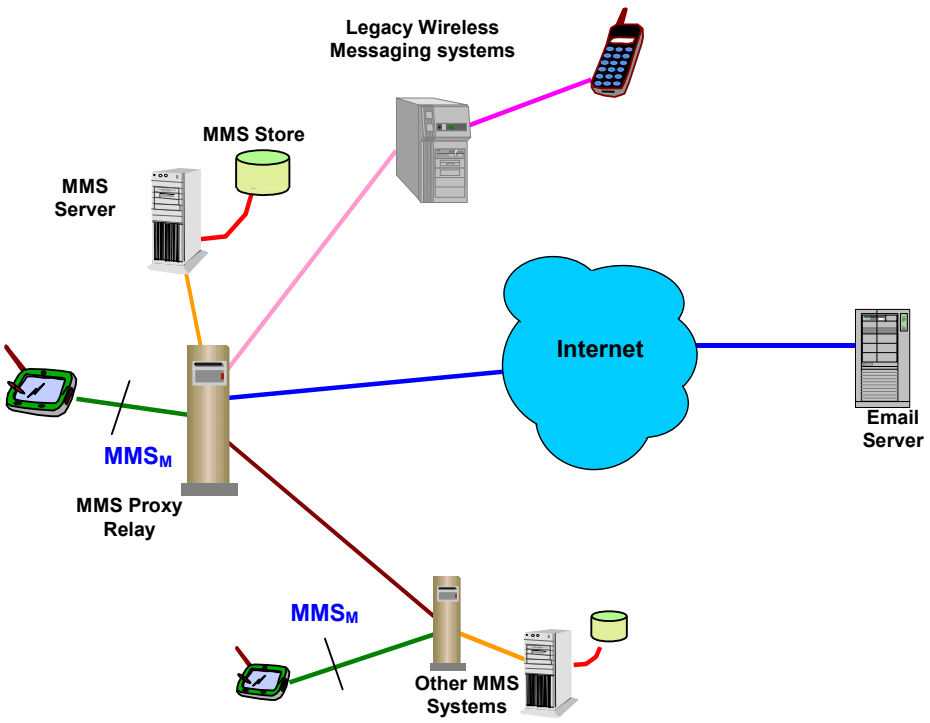

Figure 1. MMS Network Diagram with MMS Client to MMS Proxy-Relay Link Highlighted

The MMS client transactions described by this document take place on the interface labelled MMS_M in the preceding diagram.

The following figures present an amplified view of the MMS_M link. It is built on top of the WAP architecture, which permits several possible protocol stack configurations. In its role as an application, MMS provides for the delivery and services related to messaging and the data schemes that will permit presentation methods that provide for the multimedia user experience. These presentation methods are separate from MMS.

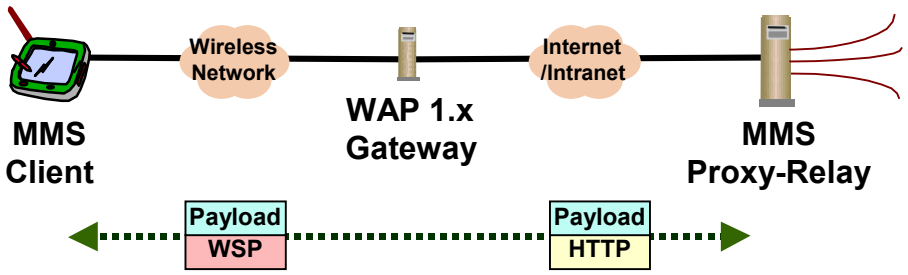

Figure 2. Implementation of MMS_M Interface Using WAP 1.x Gateway

This figure includes a few items that need to be described. The MMS Proxy-Relay is the network entity that interacts with the user mailbox and is responsible for initiating the notification process to the MMS Client. The WAP 1.x Gateway provides standard WAP services needed to implement MMS in the original WAP architecture, these include: WSP invocation of HTTP methods, see [WAPWSP]; WAP PUSH services, see [PUSHARCH]; OTA security, see [WTLS]; and, Capability Negotiations, see [UAPROF].

The above figure also shows a payload that is carried by WSP and HTTP. This payload represents the MMS application layer PDUs, which are described in the MMS Message Encapsulation [MMSENCAPS] document. It is expected that this data will be transported in its entirety between the MMS Proxy-Relay and the MMS Client.

In a different architectural configuration HTTP is used to carry MMS PDUs directly between the MMS Client and the MMS Proxy-Relay, and a gateway is only needed for push functionality. The following figure outlines such an implementation of MMS_M; note that the gateway needed for push services is omitted from the figure. Also note that a PEP may be included in the MMS_M link to provide performance enhancements, as described in [WAPARCH].

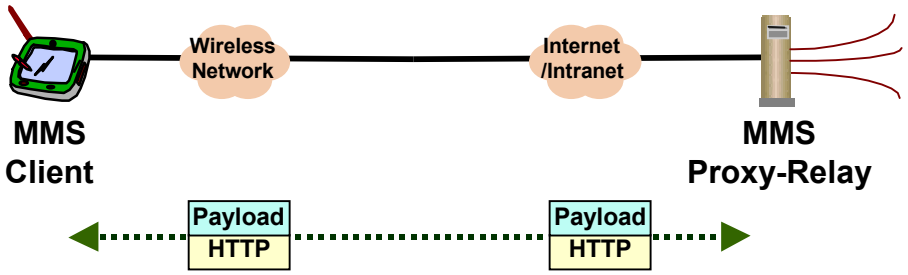

Figure 3. Implementation of MMS_M Interface Using HTTP Based Protocol Stack

The MMS application layer is the same in the different architectural configurations; the differences are contained in the two transport stacks, i.e., the WSP based protocol stack and the HTTP based protocol stack. The bindings of the MMS PDUs to these protocol stacks are defined in section 8.

This description does not address issues related to the movement or acquisition of MMs beyond the MMS Proxy-Relay as these are outside the scope of the MMS_M link.

5. Introduction to MMS Transaction Model

This section is informative.

The MMS service is realised by the invocation of transactions between the MMS Client and the MMS Proxy-Relay. These transactions include information and affect state changes on these devices. This section introduces example transaction flows and section 6 describes each individual, logically separate transaction in more detail.

The general transaction flows on MMS_M for sending, retrieving and forwarding MMs do not depend on what type of client the MM is sent to, received from, or forwarded to. The other endpoint for the MM may be another MMS Client served by the same or another MMS Proxy-Relay, it may be a client on a legacy wireless messaging system, or it may be an e-mail server.

The following four figures provide general views of the MMS_M transactions needed for: 1) an MMS Client to send an MM and receive back a resulting delivery notice; 2) an MMS Client to perform immediate retrieval of a new MM; 3) an MMS Client to perform deferred retrieval of a new MM; and, 4) an MMS Client to perform forwarding of an MM located at the MMS Proxy-Relay. The arrow labels in the following figures indicate the MMS PDUs exchanged during transactions. These PDUs are defined in detail in [MMSENCAPS].

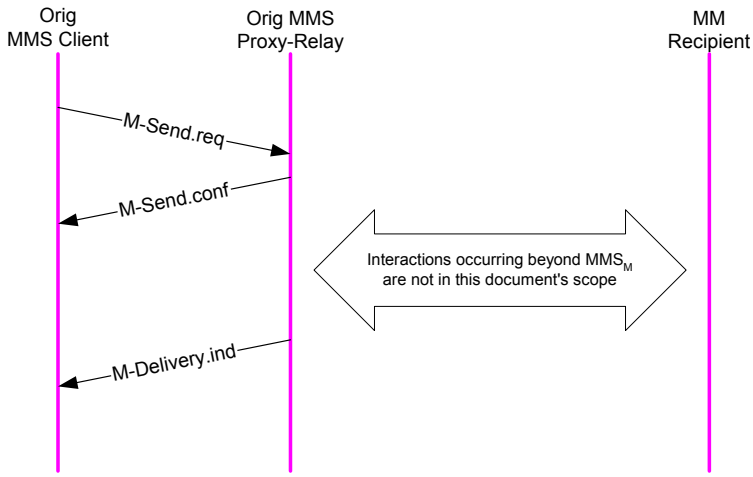

Figure 4. Example MMS_M Transaction Flow – Sending

A receiving MMS Client is said to perform immediate retrieval of a new MM when it retrieves the data from the MMS Proxy-Relay before acknowledging the message notification.

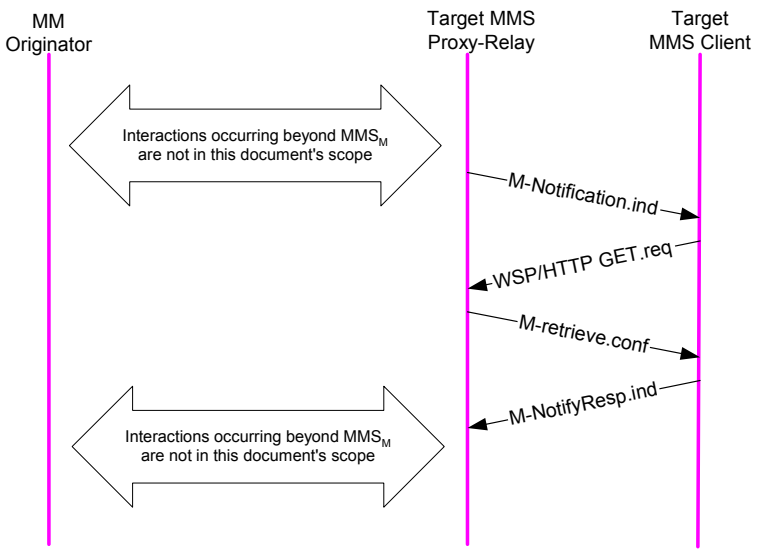

Figure 5. Example MMS$_M$ Transaction Flow – Immediate Retrieval

A receiving MMS Client is said to perform deferred retrieval of a new MM when it first acknowledges the notification and at some later point retrieves the message from the MMS Proxy-Relay.

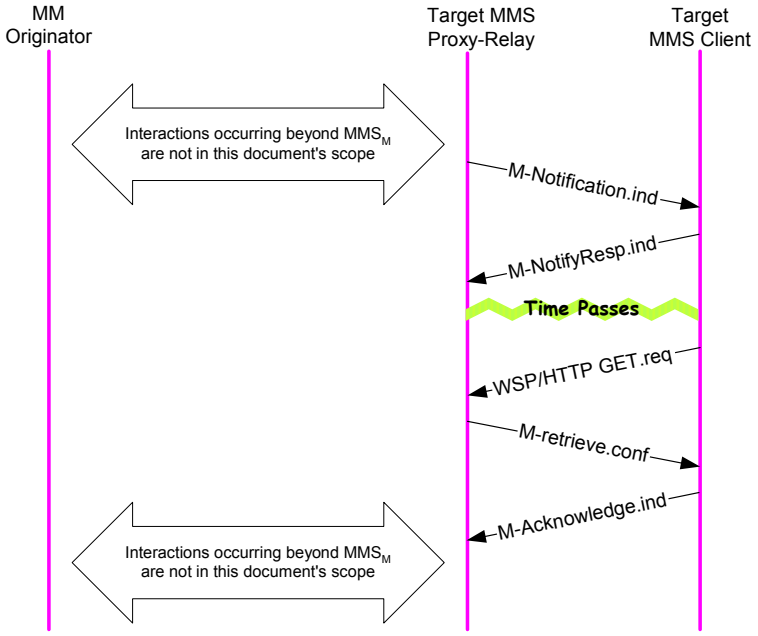

Figure 6. Example MMS$_M$ Transaction Flow – Deferred Retrieval

If both endpoints for the MM exchange are MMS Clients, the MMS$_M$ interface is involved both when the originating MMS Client sends the MM to the originating MMS Proxy-Relay and when the target MMS Client retrieves the MM from the target MMS Proxy-Relay. The following figure shows an example where both endpoints are MMS Clients and deferred retrieval is used.

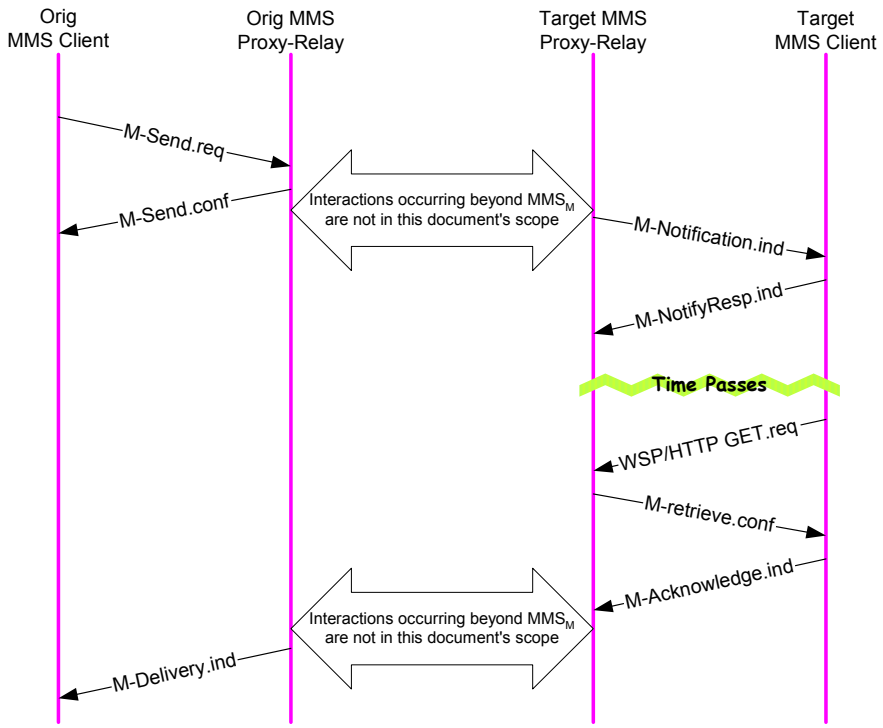

Figure 7. Example MMS Transaction Flow – Deferred Retrieval

A receiving MMS Client can request an MM located at the MMS Proxy-Relay to be forwarded to other recipients. A receiving MMS Client is said to perform deferred forwarding of an MM when it first acknowledges the notification and at some later point requests the MMS Proxy-Relay to forward the MM.

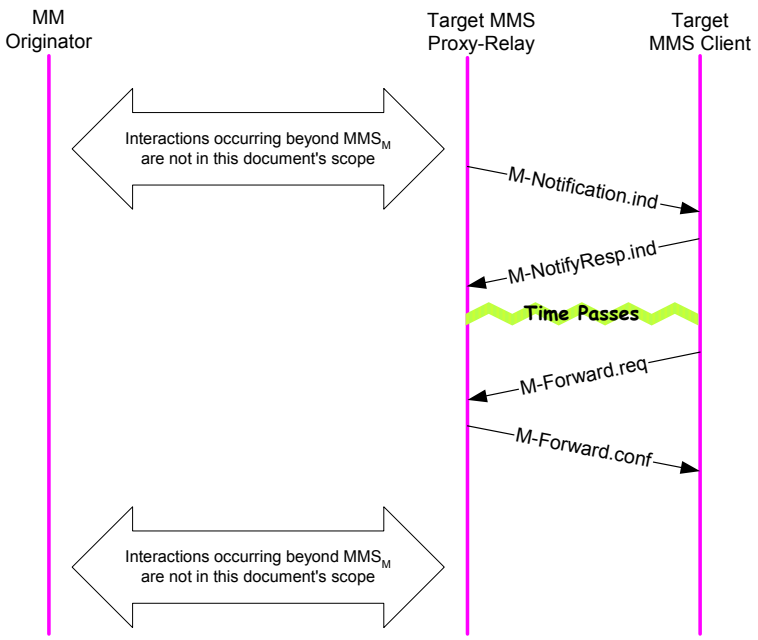

Figure 8. Example MMS_M Transaction Flow – Deferred Forwarding

A receiving MMS Client is said to perform immediate forwarding of an MM when it completes a forwarding transaction with the MMS Proxy-Relay before acknowledging the message notification. Other transaction flows further include retrieval of the MM in reaction to the notification.

As can be seen in these examples, several PDU exchanges occur on MMS_M. These PDU exchanges can be considered to form the following logically separate transactions:

- MMS Client Sending Message to MMS Proxy-Relay
- MMS Proxy-Relay Sending Notification to MMS Client
- MMS Client Retrieving Message from MMS Proxy-Relay
- MMS Client Sending Forward Request to MMS Proxy-Relay
- MMS Proxy-Relay Sending Delivery Report to MMS Client

These transactions are described in more detail in section 6.

5.1 Error Considerations

Section 6 also contains general error considerations for each transaction. For more specific information, the reader is referred to the [MMSENCAPS] and [WAPWSP] documents. The [MMSENCAPS] document also contains considerations for the case where the MMS Client and the MMS Proxy-Relay implement different versions of the MMS_M protocol described here.

5.2 Streaming Retrieval

Streaming aspects are considered outside the scope of this document and are not further discussed, except for the following paragraph.

The contents of a multimedia message may be streamed down to the recipient terminal from the MMS Proxy-Relay. The recipient MMS Proxy-Relay may convert media types and/or formats of MM contents to make it available for streaming retrieval. The decision made by the recipient MMS Proxy-Relay on whether to use streaming or not when the MMS Client retrieves the MM is based on content of the MM and/or user settings/preferences and/or UAProf [UAPROF] capability negotiation and/or the capabilities of the MMS Proxy-Relay. If streaming retrieval is used, the streaming-specific protocols, codecs, presentation, file format, media adaptation, control, and session negotiation may be according to [PSSGEN], [PSSCOD] and [3G2VID].

6. MMS Client Transactions

This section is normative. It describes MMS transactions at the application layer. For normative information about MMS PDU binding to the underlying transport, please refer to section 8. The PDUs and header fields referred to in the following SHALL comply with the definitions in [MMSENCAPS].

6.1 MMS Client Sending Message to MMS Proxy-Relay

The process for a client to send a message is built on top of the **M-Send** transaction. It provides the mechanism for the MMS Client to submit an MM to the MMS Proxy-Relay and to get back information in response. The following Figure 9 gives an example of this transaction.

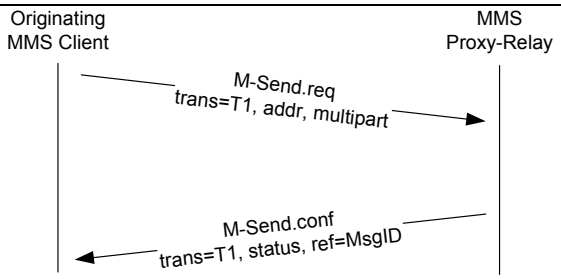

Figure 9. Example MMS Send Transaction

6.1.1 Transaction Flow

The MMS Client that wishes to send an MM SHALL include the MM in an **M-Send.req** and submit the resulting PDU to the MMS Proxy-Relay that supports the specific MMS Client.

The MMS Client SHALL compose a transaction ID for the submitted PDU. This transaction ID is used by the MMS Client and MMS Proxy-Relay to provide linkage between the originated **M-Send.req** and the response **M-Send.conf** PDUs. The value used for the transaction ID is determined by the MMS Client and no interpretation is expected by the MMS Proxy-Relay.

Upon receipt of the **M-Send.req** PDU, the MMS Proxy-Relay SHALL respond with an **M-Send.conf** PDU. This response PDU SHALL provide a status code for the requested operation. If the MMS Proxy-Relay is willing to accept the request to send the message, the status SHALL be '**ok**' and the PDU SHALL include a message-ID that MAY be used for following activities that need to refer to the specific sent MM (e.g. delivery reports).

6.1.2 Error Considerations

Various error cases may exist. These include network faults, server faults and service faults. For network faults (e.g. server not available) or server faults (e.g. bad path) the MMS Client SHALL receive an error indication that relates to the transport layer error that was detected. These errors MAY be recoverable (e.g. MMS Proxy-Relay down temporarily) or may be more permanent in nature. Strategies for recovery or retry are beyond the scope of this document to address.

Service errors are different. In these cases the MMS Proxy-Relay actually received the **M-Send.req** PDU and responds with an **M-Send.conf** PDU with the appropriate error code.

6.2 MMS Proxy-Relay Sending Notification to MMS Client

To inform an MMS Client that an MM is available and for it to return back information, a set of asynchronous PDUs, **M-Notification.ind** and **M-NotifyResp.ind**, are utilized. This provides the mechanism for the MMS Proxy-Relay to notify the MMS Client with certain factors about the new MM. This will let the MMS Client retrieve the MM.

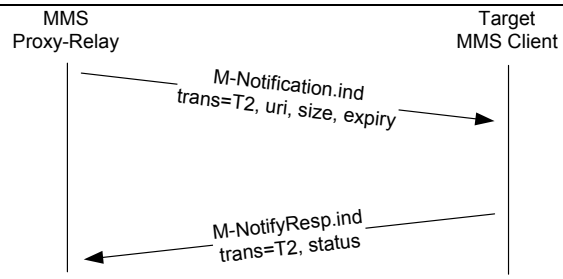

Figure 10. Example MMS Notification Transaction

6.2.1 Transaction Flow

The MMS Proxy-Relay SHALL utilise the **M-Notification.ind** PDU when it needs to inform the MMS Client that a message is available for delivery.

The information conveyed SHALL include an [RFC2396] compliant URI that will be used to actually retrieve the MM in a subsequent operation by the MMS Client. Additional information about the message (e.g. message size, expiry) may be used by the MMS Client to determine its behaviour. For example, the MMS Client may defer the retrieval of the MM until after a user confirmation if it exceeds a size threshold.

The MMS Proxy-Relay SHALL compose a transaction ID for the notification PDU. This transaction ID is used by the MMS Client and MMS Proxy-Relay to provide linkage between the originated **M-Notification.ind** and the response **M-NotifyResp.ind** PDUs. The value used for the transaction ID is determined by the MMS Proxy-Relay and no interpretation is expected by the MMS Client.

Upon receipt of the **M-Notification.ind** PDU, the MMS Client SHALL respond with an **M-NotifyResp.ind** PDU to the MMS Proxy-Relay that supports the specific MMS Client.

The **M-NotifyResp.ind** response PDU SHALL provide a message retrieval status code. The status '**retrieved**' SHALL be used only if the MMS Client has successfully retrieved the MM prior to sending the **NotifyResp.ind** response PDU.

6.3 MMS Client Retrieving Message from MMS Proxy-Relay

The operation for retrieval of the MM by the MMS Client from the MMS Proxy-Relay is built upon the normal WSP/HTTP **GET** functionality. Therefore, no new operation is actually defined. The message type for the PDU returned from the MMS Proxy-Relay to the MMS Client is **M-retrieve.conf**.

Delivery of the MM MAY be either before or after the **M-NotifyResp.ind** PDU, depending on immediate retrieval or deferred retrieval of MM respectively. The MMS Proxy-Relay MAY therefore decide to request an acknowledgement from the MMS Client to confirm successful retrieval in case of deferred retrieval. These variations are shown in Figure 11 and Figure 12 respectively.

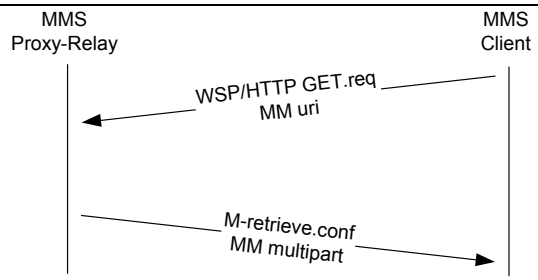

Figure 11. Example MMS Retrieval Transaction without Acknowledgement

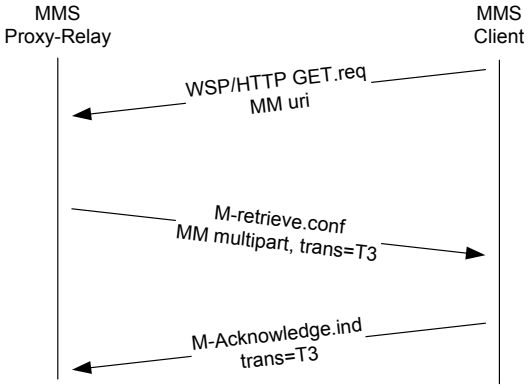

Figure 12. Example MMS Retrieval Transaction with Acknowledgement

6.3.1 Transaction Flow

The MMS Client SHALL initiate the retrieval activity by utilizing the URI that was delivered to it in the **M-Notification.ind** PDU using the normal WSP/HTTP **GET** method operation.

The response PDU **M-retrieve.conf**, if successful, contains the MM. This PDU SHALL include MMS header fields providing additional information. This response PDU MAY provide a status code for the requested retrieval operation.

Depending on the MMS Proxy-Relay needs, the **M-retrieve.conf** response that it provides MAY request an acknowledgement to be generated by the MMS Client. The MMS Proxy-Relay MAY make this request based on whether or not it needs to provide a delivery notice back to the originator of the MM. Alternatively, it MAY make that request based upon an expectation that it would then be able to delete the MM from its own store. This decision is not a part of this transaction.

The MMS Proxy-Relay SHALL make this request for acknowledgement by including a transaction ID in the **M-retrieve.conf** PDU. This transaction ID is used by the MMS Client and MMS Proxy-Relay to provide linkage between the originated **M-retrieve.conf** and the response **M-Acknowledge.ind** PDUs. The value used for the transaction ID is determined by the MMS Proxy-Relay and no interpretation is expected by the MMS Client.

If an acknowledgement is requested, the MMS Client SHALL respond with an **M-Acknowledge.ind** PDU to the MMS Proxy-Relay that supports the specific MMS Client. The **M-Acknowledge.ind** PDU confirms successful message retrieval to the MMS Proxy Relay.

6.3.2 Error Considerations

If the URI can not be resolved, a network or server fault MAY be returned. For example, if the MMS Server deletes the MM from the store, making the requested MM unavailable, it is expected that the WSP/HTTP request will generate a 'Data Not Available' status code (e.g. 404). In this case, the lower level error would be returned.

Service/application level errors are different. In these cases, the MMS Proxy-Relay actually received the WSP/HTTP **GET** message and responds with an **M-retrieve.conf** PDU with the appropriate error code. The MMS Proxy-Relay SHALL not include the transaction ID in the **M-retrieve.conf** PDU for these error cases.

6.3.3 Clarifications

To some readers it may not appear consistent that the WSP/HTTP GET.req message is shown in Figure 11 and Figure 12 as this message belongs to a different protocol layer than the MMS PDUs. However, the figures are consistent in that they define the MMS Client Transactions in terms of the packets sent "across the wire" between peer entities (and not in terms of primitives, which are defined between layers in a protocol stack). The appearance of WSP/HTTP GET.req in the diagrams is not to be taken as a recommendation to bypass the implementation of layered protocols.

6.4 MMS Client Sending Forward Request to MMS Proxy-Relay

The **M-Forward** transaction enables an MMS Client to request forwarding of an MM located at the MMS Proxy-Relay and to get back information in response. The following figure gives an example of this transaction.

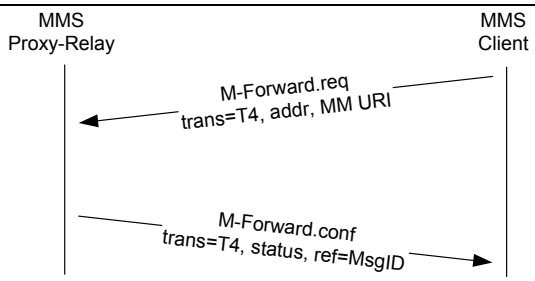

Figure 13. Example MMS Forward Transaction

6.4.1 Transaction Flow

Support of the MMS forward transaction is OPTIONAL for MMS Proxy-Relay and MMS Client. The following description applies when both entities support the forward transaction.

The MMS Client that intends to forward an MM located at the MMS Proxy-Relay SHALL send an **M-Forward.req** PDU to the MMS Proxy-Relay that supports the specific MMS Client. The message SHALL contain the URI of the MM to be forwarded which was conveyed to the MMS Client as part of the **M-Notification.ind** PDU. Additionally the **M-Forward.req** PDU SHALL contain the address of one or more recipients that the MM should be forwarded to.

The MMS Client SHALL compose a transaction ID and include it in the **M-Forward.req** PDU. This transaction ID is used by the MMS Client and MMS Proxy-Relay to provide linkage between the originated **M-Forward.req** PDU and the corresponding **M-Forward.conf** PDU. The value used for the transaction ID is determined by the MMS Client and no interpretation is expected by the MMS Proxy-Relay.

After reception of the **M-Forward.req** PDU, the MMS Proxy-Relay SHALL respond with an **M-Forward.conf** PDU. This response PDU SHALL provide a status code for the requested operation. If the MMS Proxy-Relay is willing to accept the request to forward the MM, the status SHALL be '**ok**' and the PDU SHALL include a message-ID that MAY be used for following activities that need to refer to the specific message forwarded (e.g. delivery reports).

The MMS forward transaction is logically independent to both the MMS notification transaction and the MMS retrieval transaction, i.e., the MMS Client SHALL react to an **M-Notification.ind** PDU by sending an **M-NotifyResp.ind** PDU before, during or after an optional MMS forward transaction. Furthermore the MMS Client MAY also react to an **M-Notification.ind** PDU by initiating a retrieval operation. This additional operation on the MM may be supported by the MMS Client and/or the MMS Proxy-Relay.

6.4.2 Error Considerations

Various error cases may exist. These include network faults, server faults and service faults. For network faults (e.g. server not available) or server faults (e.g. bad path) the MMS Client SHALL receive an error indication that relates to the transport layer error that was detected. These errors MAY be recoverable (e.g. MMS Proxy-Relay down temporarily) or may be more permanent in nature. Strategies for recovery or retry are beyond the scope of this document to address.

Service errors are different. In these cases the MMS Proxy-Relay actually received the **M-Forward.req** PDU and responds with either an **M-Forward.conf** PDU if it does support MMS forward transactions, or an **M-Send.conf PDU** if it does not support MMS forward transactions; in either case the appropriate error code SHALL be included in the response.

6.5 MMS Proxy-Relay Sending Delivery Report to MMS Client

To permit the originating MMS Client to know when a message delivery has occurred the **M-Delivery.ind** PDU has been defined to provide that information. The **M-Delivery.ind** PDU originates at the MMS Proxy-Relay providing information to the MMS Client about the MM that was delivered. There is no associated response or acknowledgment PDU. The following Figure 14 shows an example of this transaction.

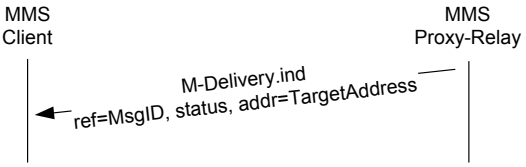

Figure 14. Example Delivery Report

6.5.1 Transaction Flow

The **M-Delivery.ind** PDU SHALL be sent by the MMS Proxy-Relay to the MMS Client to convey information about the status of a particular MM delivery that was performed. The MM is identified by the message-ID that was generated when the original message was submitted or forwarded. It also provides addressing information of the originally targeted entity.

If an MM was addressed to multiple entities, multiple **M-Delivery.ind** PDUs SHOULD be expected to be returned, one for each addressed entity.

6.5.2 Error Considerations

The **M-Delivery.ind** PDU is generated when the MMS Proxy-Relay is satisfied that it has sufficient information to declare that the MM was delivered or other status can be declared. As such, there may be cases where the MMS Proxy-Relay makes a decision about the delivery status that may be incorrect (e.g. timer expiry may generate an expiry notice but target MMS Client may actually retrieve MM if the read occurred before the MM was deleted).

There is no associated response or acknowledgment PDU defined for the **M-Delivery.ind** PDU. The success rate for transmittal of the **M-Delivery.ind** PDU is dependent upon the quality of service provided by the transport service(s) utilized.

6.5.3 Other Issues

A target MMS Client may, within an **M-NotifyResp.ind** PDU or an **M-Acknowledge.ind** PDU, request denial of an originator's request for delivery notification. Therefore, an MMS Client SHOULD NOT expect to receive all the delivery reports that it may have requested.

6.6 Read Reports

There are two ways of handling Read Reports, either in the form of an MM or in the form of a PDU designed for that purpose. If the MMS Client is of higher version than 1.0 and if it supports read reporting it MUST also support the handling of Read Report PDUs.

6.6.1 Multimedia Message Read Report

When the originating MMS Client requests a Read Report for an MM, the receiving MMS Client MAY send a Read Report back to it. This report MAY be sent and delivered as a regular MM using the mechanisms described in this section.

To permit a user to determine that an MM is a Read Report, a few fields can be used to provide that information:

- The subject field should be copied from the original, prepending a 'Read:' to the text.
- The Message-ID of the original message is available and should be included in the message body.
- The body of the message may provide information about the read action or status.

The following is an example of a Read Report MM. It is in response to an MM that user A had sent to user B:

```
From:    B
To:      A
Sent:    Friday, January 21, 2000 1:50 PM
Subject: Read: My Message
Your message

    To:          B
    Subject:     My Message
    Message-ID:  <200002211806.MAA26265@mail1.domain.com>
    Sent:        1/21/2000 1:29 PM

was read on 1/21/2000 1:50 PM.
```

6.6.2 PDU Read Report

When the originating MMS Client requests a Read Report for an MM, the receiving MMS Client MAY send a Read Report back to it. This report MAY be sent in the form of a PDU and delivered using the mechanisms described in this section.

The **M-read-rec.ind** PDU originates at the recipient MMS Client and is sent via the recipient MMS Proxy-Relay to the originating MMS Proxy-Relay. Upon receiving the **M-read-rec.ind**, the originating MMS Proxy-Relay forwards an **M-read-orig.ind** PDU to the originating MMS Client. The following Figure 15 and Figure 16 outline the transaction flow.

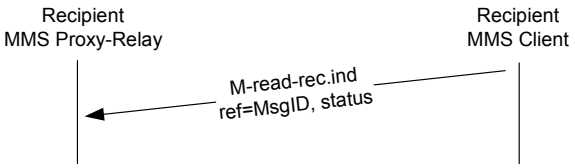

Figure 15. Read Report transaction flow on the recipient side

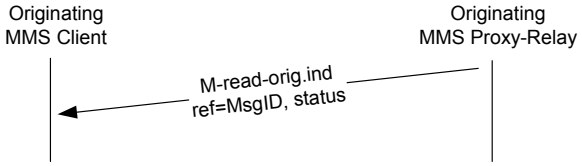

Figure 16. Read Report transaction flow on the originating side

6.6.3 Transaction Flow

If supported by a receiving MMS Client a Read Report to an MM marked with the **Read-Report** flag MAY be sent in the form of a new MM, in the form of a PDU or not sent at all.

6.6.3.1 Multimedia Message Read Report Transaction Flow

When the Read Report is sent in the form of a regular MM the report SHALL be sent using the normal `M-Send` operation as it is just another MM origination. As such, it SHALL be delivered using the normal delivery methods. Due to the nature of the MM, the `Message-Class` header field SHALL have the value '`Auto`', the `Read-Report` flag MUST NOT be set, and the `Delivery-Report` flag MUST NOT be set in a Read Report MM.

The MMS Client receiving a Read Report MM will see it as a new MM. The interpretation as a Read Report is done by context. In cases where the original MM had multiple addresses, the MMS Client SHOULD expect that multiple Read Reports will be returned.

6.6.3.2 PDU Read Report Transaction Flow

When the Read Report is handled by PDUs and the recipient MMS Client wishes to send a Read Report a **M-read-rec.ind** MUST be sent to the recipient Proxy-Relay from the recipient MMS Client. The recipient Proxy-Relay then forwards the **M-read-rec.ind** information to the originating Proxy-Relay, that transaction is however out of scope for the WAP Forum specifications.

To handle backward compatibility, the originating Proxy-Relay MAY keep a record of what version of **M-Send.req** the originating MMS Client used for submitting the corresponding MM. If that version number is 1.0 the originating Proxy-Relay MAY transform the **M-read-rec.ind** information into an MM. If a Read Report is sent as an MM, it MUST follow the rules described in sections 6.6.1 and 6.6.3.1.

If the originating Proxy-Relay is aware of that the originating MMS Client is of higher version than 1.0 or is unaware of the MMS Client version, it MUST forward a Read Report in the form of a **M-read-orig.ind** PDU upon receiving the **M-read-rec.ind** information.

In cases where the original message had multiple addresses, the MMS Client SHOULD expect that multiple Read Report PDUs will be returned.

6.6.4 Error Considerations

Origination and delivery of a Read Report MM is as for a normal MM and does not require additional error considerations.

6.6.5 Other Issues

Since the handling of Read Reports is an optional capability of the recipient MMS Client and the recipient MMS Client may in any case choose not to send a Read Report, an originating MMS Client SHOULD NOT depend upon receiving a Read Report in all cases.

If the originating MMS Proxy-Relay does not track the version number of the originating MMS Client , the originating MMS Client may not receive a Read Report it can understand. In this case the MMS Client shall notify the MMS Proxy-Relay according to the regular procedures used when an MMS Client receives a PDU it does not understand (as described in section 6.8 of [MMSENCAPS]). Upon receiving this compatibility problem information, the MMS Proxy-Relay MAY create an MM Read Report as described in sections 6.6.1 and 6.6.3.1 and send it to the originating MMS Client.

Due to backward compatibility a Read Report may always come in the form of an MM since there is no mandated or standardized function to transform a Read Report MM into a Read Report PDU.

6.7 Security Considerations

At present, the end-to-end security aspects of the MMS_M PDUs are dependent upon the security provided by the transport service(s) utilized. The use of MMS with transport layer hop-by-hop security protocols is described in section 8.3.2.

When the MMS Proxy-Relay in an M-Notification.ind PDU identifies a new MM to the MMS Client using the https URI scheme, the MMS Client SHALL use a secure transport for a subsequent retrieval operation, or indicate an error to the user.

7. Terminal Capability Negotiation

This section is normative.

If the MMS Client performs capability negotiation then it MUST use the mechanism specified in [UAPROF]. The MMS Proxy Relay SHOULD support this mechanism.

If using capability negotiation, the MMS Client SHALL indicate its capabilities within the UAProf information by using attributes from the MMS Characteristics component defined below and OPTIONALLY by using attributes from other components of the UAProf schema. The MMS Proxy-Relay SHOULD use this information in preparation of messages to be delivered to the MMS Client.

The MMS Proxy-Relay MAY adjust an MM to be delivered that contains media types that are not supported by the MMS Client. This adjustment MAY involve the deletion or adaptation of those unsupported media types.

7.1 MMS attributes in other components of the UAProf schema

This section is informative.

The UAProf specification includes a schema containing attributes that describe the client hardware, the browser user-agent, network characteristics and more. Some of the attributes included in the aforementioned specification also apply to the MMS Client, e.g. “ScreenSize”, “CpuType”, and “PushMessageSize”. For a complete reference to the attributes available in the UAProf schema, please see [UAPROF].

7.2 Summary of the MMS Characteristics component

This section is informative. A normative description can be found in Appendix A.1.

The table below summarizes the attributes defined within the MMS Characteristics component.

Attribute	Description	Resolution Rule	Type	Sample Values
Component: MmsCharacteristics				
MmsMaxMessageSize	The maximum size of a multimedia message in bytes.	Locked	Number	20480
MmsMaxImageResolution	The maximum size of an image in units of pixels (horizontal x vertical).	Locked	Literal	"80x60"

MmsCcppAccept	List of supported content types conveyed as MIME types.	Locked	Literal bag	“image/jpeg”, “audio/wav”, “video/mpeg-4”
MmsCcppAcceptCharSet	List of character sets that the MMS Client supports. Each item in the list is a character set name registered with IANA.	Locked	Literal bag	“US-ASCII”, “ISO-8859-1”
MmsCcppAcceptLanguage	List of preferred languages. The first item in the list should be considered the user’s first choice. Property value is a list of natural languages, where each item in the list is the name of a language as defined by [RFC1766].	Locked	Literal bag	“en”, “fr”
MmsCcppAcceptEncoding	List of transfer encodings that the MMS Client supports. Property value is a list of transfer encodings, where each item in the list is a transfer encoding name as specified by [RFC2045] and registered with IANA.	Locked	Literal bag	“base64”, “quoted-printable”
MmsVersion	The MMS versions supported by the MMS Client conveyed as majorVersionNumber.minorVersionNumber.	Locked	Literal bag	"2.0", "1.3"
MmsCcppStreamingCapable	Indicates whether the MMS Client is capable of invoking streaming.	Locked	Boolean	"Yes", "No"

8. Binding to Transport Protocols

This section is normative; it defines how the MMS PDUs can be conveyed over HTTP, WSP and WAP Push. The structures of the MMS PDUs are defined in [MMSENCAPS].

The HTTP/WSP response codes reflect only HTTP/WSP layer conditions. MMS is defined at the application layer; MMS status codes and error conditions are handled through the MMS PDUs.

The Content-Type header defined in [RFC2616], [WSP] and [PUSHMSG] MUST be used with the MMS PDUs. The content type of the PDUs is application/vnd.wap.mms-message.

When setting up a WSP session and when sending WSP/HTTP GET requests the MMS Client SHOULD convey the capabilities of the terminal and of the MMS Client. For more details see section 7 on Terminal Capability Negotiation.

8.1 Binding When MMS Client Uses WSP Based Stack

The table below gives an overview of how MMS PDUs are bound to the WSP based protocol stack and the HTTP based protocol stack when MMS_M is implemented using a WAP 1.x Gateway.

PDU	Client ↔ Proxy-Relay	Client ↔ WAP 1.x Gateway/PPG	WAP 1.x Gateway/PPG ↔ Proxy-Relay
M-Send.req	→	WSP POST	HTTP POST
M-Send.conf	←	WSP 200 OK	HTTP 200 OK
M-Retrieve.conf	←	WSP 200 OK	HTTP 200 OK
M-Notification.ind	←	Confirmed Push or Unconfirmed Push	Push Access Protocol
M-NotifyResp.ind	→	WSP POST	HTTP POST
M-Delivery.ind	←	Confirmed Push or Unconfirmed Push	Push Access Protocol
M-Acknowledge.ind	→	WSP POST	HTTP POST
M-Read-Rec.ind	→	WSP POST	HTTP POST
M-Read-Orig.ind	←	Confirmed Push or Unconfirmed Push	Push Access Protocol
M-Forward.req	→	WSP POST	HTTP POST
M-Forward.conf	←	WSP 200 OK	HTTP 200 OK

Table 1: MMS PDU Bindings When MMS Client Uses WSP Based Stack

WSP [WAPWSP] provides both a connection-mode session service and a connectionless session service. The MMS Client may utilize either of these when invoking a WSP request operation. If using the connection-mode service, then, unless there is an existing WSP session between the WAP 1.x Gateway and the terminal, one has to be set up before the sending of a WSP request.

For more details about the transport layer bindings when the MMS Client uses the WSP based protocol stack, please refer to section 8.1.1 through 8.1.6.

8.1.1 Sending an MM

To submit an MM, the MMS Client SHALL invoke a WSP **POST** operation with the **M-Send.req** PDU embedded as the content body. This **POST** is submitted using a URI that addresses the MMS Proxy-Relay that supports the specific MMS Client. The WAP 1.x Gateway then forwards the PDU to the MMS Proxy-Relay using regular HTTP **POST** [RFC2616].

The '**200 OK**' response to the WSP/HTTP **POST** that contained the **M-Send.req** PDU carries the confirmation PDU, i.e., the **M-Send.conf**.

8.1.2 Notification

The **M-Notification.ind** is carried as push content, either over confirmed or unconfirmed push [PUSHOTA]. The Push Initiator, i.e., the MMS Proxy-Relay, requests the type of push to be used [PUSHPAP] when submitting the **M-Notification.ind** to the PPG.

The **M-Notification.ind** PDU SHALL be sent as the message body of a [PUSHMSG]. The X-Wap-Application-Id message header of that push message MUST be set to 'x-wap-application:mms.ua' if the absoluteURI form of the app-id syntax is used, and MUST be set to '4' if the app-assigned-code form of the app-id syntax is used.

To send a notification response, the MMS Client SHALL invoke a WSP **POST** operation with an **M-NotifyResp.ind** PDU embedded as the content body. This **POST** is submitted using a URI that addresses the MMS Proxy-Relay that supports the specific MMS Client. The WAP 1.x Gateway then forwards the PDU to the MMS Proxy-Relay using regular HTTP **POST** [RFC2616]. The MMS Client SHOULD ignore the associated WSP/HTTP **POST** response from the MMS Proxy-Relay.

8.1.3 Retrieving an MM

The terminal invokes a WSP **GET** operation. The '**200 OK**' response to the **GET** contains the **M-Retrieve.conf** PDU, which includes the actual MM.

The **M-Acknowledge.ind** PDU is transported in the same way as the **M-NotifyResp.ind** PDU, i.e., by a WSP **POST** between the terminal and the WAP 1.x Gateway and then forwarded via HTTP to the MMS Proxy-Relay. The MMS Client SHOULD ignore the associated WSP/HTTP **POST** response from the MMS Proxy-Relay.

8.1.4 Delivery Report

The **M-Delivery.ind** is pushed from the originating MMS Proxy-Relay to the originating terminal the same way as the **M-Notification.ind** on the receiving side.

8.1.5 Read Report

The **M-Read-Rec.ind** PDU is transported in the same way as the **M-NotifyResp.ind** PDU. The **M-Read-Orig.ind** PDU is transported in the same way as the **M-Notification.ind** PDU.

8.1.6 Forward Without Download

The **M-Forward.req** is transported in the same way as the **M-Send.req**. The **M-Forward.conf** is transported in the same way as the **M-Send.conf**.

8.2 Binding When MMS Client Uses HTTP Based Stack

The table below gives an overview of how MMS PDUs are bound to the HTTP based protocol stack when the MMS_M link does not employ a WAP 1.x Gateway.

PDU	Client ↔ Proxy-Relay	Method
M-Send.req	→	HTTP POST
M-Send.conf	←	HTTP 200 OK
M-Retrieve.conf	←	HTTP 200 OK
M-Notification.ind	←	Push Access Protocol between MMS Proxy-Relay and PPG; Confirmed Push or Unconfirmed Push between PPG and MMS Client
M-NotifyResp.ind	→	HTTP POST
M-Delivery.ind	←	Push Access Protocol between MMS Proxy-Relay and PPG; Confirmed Push or Unconfirmed Push between PPG and MMS Client
M-Acknowledge.ind	→	HTTP POST
M-Read-Rec.ind	→	HTTP POST
M-Read-Orig.ind	←	Push Access Protocol between MMS Proxy-Relay and PPG; Confirmed Push or Unconfirmed Push between PPG and MMS Client
M-Forward.req	→	HTTP POST
M-Forward.conf	←	HTTP 200 OK

Table 2: MMS PDU Bindings When MMS Client Uses HTTP Based Stack

For more details about the transport layer bindings when the MMS Client uses the HTTP based protocol stack, please refer to section 8.2.1 through 8.2.6.

8.2.1 Sending an MM

Unless there is an existing TCP connection between the MMS Proxy-Relay and the terminal, one has to be set up before the sending of an MM [WP-HTTP]. To submit an MM, the MMS Client SHALL invoke an HTTP **POST** operation with the **M-Send.req** PDU embedded as the content body. This **POST** is submitted using a URI that addresses the MMS Proxy-Relay that supports the specific MMS Client.

The '**200 OK**' response to the HTTP **POST** that contained the **M-Send.req** PDU carries the confirmation PDU, i.e., the **M-Send.conf**.

8.2.2 Notification

The **M-Notification.ind** is carried as push content, either over confirmed or unconfirmed push [PUSHOTA]. The Push Initiator, i.e., the MMS Proxy-Relay, requests the type of push to be used [PUSHPAP] when submitting the **M-Notification.ind** to the PPG.

The **M-Notification.ind** PDU SHALL be sent as the message body of a [PUSHMSG]. The X-Wap-Application-Id message header of that push message MUST be set to 'x-wap-application:mms.ua' if the absoluteURI form of the app-id syntax is used, and MUST be set to '4' if the app-assigned-code form of the app-id syntax is used.

To send a notification response, the MMS Client SHALL invoke an HTTP **POST** operation with an **M-NotifyResp.ind** PDU embedded as the content body. This **POST** is submitted using a URI that addresses the MMS Proxy-Relay that supports the specific MMS Client. The MMS Client SHOULD ignore the associated HTTP **POST** response from the MMS Proxy-Relay.

8.2.3 Retrieving an MM

If there is no existing TCP connection one has to be set up before the retrieval of an MM. The terminal uses the established TCP connection to do an HTTP **GET** operation. The '**200 OK**' response to the **GET** contains the **M-Retrieve.conf** PDU, which includes the actual MM.

The **M-Acknowledge.ind** PDU is transported in the same way as the **M-NotifyResp.ind** PDU, i.e., by an HTTP **POST** over a TCP connection between the terminal and the MMS Proxy-Relay. The MMS Client SHOULD ignore the associated HTTP **POST** response from the MMS Proxy-Relay.

8.2.4 Delivery Report

The **M-Delivery.ind** is pushed from the originating MMS Proxy-Relay to the originating terminal the same way as the **M-Notification.ind** on the receiving side.

8.2.5 Read Report

The **M-Read-Rec.ind** PDU is transported in the same way as the **M-NotifyResp.ind** PDU. The **M-Read-Orig.ind** PDU is transported in the same way as the **M-Notification.ind** PDU.

8.2.6 Forward Without Download

The **M-Forward.req** is transported in the same way as the **M-Send.req**. The **M-Forward.conf** is transported in the same way as the **M-Send.conf**.

8.3 URI Schemes

Every WSP request and every HTTP request includes a URI. The following standard URI schemes are defined for MMS Clients.

8.3.1 The http URI Scheme

The http URI scheme identifies a particular URI syntax suitable for naming resources stored on hypermedia origin servers. The specification of an http scheme does not imply the use of a particular communication protocol between an MMS Client and an MMS Proxy-Relay or a WAP 1.x Gateway. An MMS Client SHALL access the MMS Proxy-Relay specified by the URI either directly using WP-HTTP/HTTP, or via a WAP 1.x Gateway that supports protocol conversion between HTTP and WSP. Alternatively the URI may specify an MMS Proxy-Relay, which combines the function of a WAP 1.x Gateway and MMS Proxy-Relay into one entity. In this case the resource SHALL be accessed directly across the WSP protocol.

8.3.2 The https URI Scheme

The https URI scheme indicates that the named resource is on a secure origin server. This conventionally implies the use of HTTP [RFC2616] over a transport layer security protocol such as TLS between the client and the origin server.

An MMS Client accessing the resource specified by the https URI using the connection-oriented protocol stack, consisting of Wireless Profiled HTTP [WP-HTTP] and Wireless Profiled TCP [WP-TCP], MUST use Wireless Profiled HTTP over the transport layer security protocol in accordance with "WAP TLS Profile and Tunnelling Specification" [WP-TLS] to communicate with the MMS Proxy-Relay; otherwise, an indication of lack of security MUST be given to the user and the operation MUST be aborted.

If an MMS Client uses the connection-oriented protocol through a PEP to communicate with the MMS Proxy-Relay, the PEP behaves in accordance with [RFC2817] and tunnels the transport layer security protocol (e.g., WP-TLS) through the PEP for the duration of the secure session. The MMS Client MUST use the HTTP CONNECT method [WP-TLS] to establish the secure transport protocol tunnel with the MMS Proxy-Relay. An indication MUST be given to the user and the operation MUST be aborted if a tunnel cannot be established.

The establishment of a secure session using the connection-oriented protocol SHOULD be indicated to the user. This applies whether the secure session operates directly or through one or more PEPs.

An MMS Client accessing the resource specified by the https URI using the datagram-oriented protocol stack, consisting of [WDP], [WTP] + WSP with WTLS providing the transport layer security protocol, MUST request the resource via [WAPWSP] using a [WTLS] secured session. An indication MUST be given to the user and the operation MUST be aborted if the WTLS session cannot be established.

If the MMS Client accessing the resource specified by the https URI uses the datagram-oriented protocol through a WAP 1.x Gateway to communicate to the MMS Proxy-Relay, the WAP 1.x Gateway uses HTTP/1.1 [RFC2616] over TLS or SSL to communicate with the MMS Proxy-Relay; otherwise, the WAP 1.x Gateway denies the operation and responds with an indication to the MMS Client by a well known warning or error code. In order to indicate to the MMS Proxy-Relay that the transport layer security consists of WTLS between MMS Client and WAP 1.x Gateway, the WAP 1.x Gateway inserts the

HTTP VIA header with the received-protocol for the secure WSP connection being indicated as WSPS/<WSP version number> to indicate secure WSP.

Appendix A. Static Conformance Requirements (Normative)

The format, contents and syntax of the tables in this section are as mandated by [CREQ].

The SCR items in the "Requirement" column can be found in the following WAP specifications:

- SCR items starting with "UAProf" are defined in [UAPROF].
- SCR items starting with "MMSE" are defined in [MMSENCAPS].
- SCR items starting with "MSG" are defined in [PUSHMSG].
- SCR items starting with "OTA" are defined in [PUSHOTA].
- SCR items starting with "WSP" are defined in [WAPWSP].
- SCR items starting with "HTTP" are defined in [WP-HTTP].
- SCR items starting with "TLS" are defined in [WP-TLS].
- SCR items starting with "WTLS" are defined in [WTLS].

A.1 MMS Client

A.1.1 Client Level Function Groups

Item	Function	Reference	Status	Requirement
MMSCTR-CLF-C-001	Support for MMS Client Functions	7	M	MMSCTR-CLF-C-002 OR MMSCTR-CLF-C-003
MMSCTR-CLF-C-002	Support for Originating MMS Client Functions	7.1, 7.5	O	MMSCTR-SND-C-001 AND MMSCTR-DRP-C-001
MMSCTR-CLF-C-003	Support for Receiving MMS Client Functions	7.2, 7.3	O	MMSCTR-NTF-C-001 AND MMSCTR-FTC-C-001
MMSCTR-CLF-C-004	Capability Negotiation between Receiving MMS Client and MMS Proxy-Relay using the UAProf Component MMS Characteristics	8, 9	O	MMSCTR-CLF-C-003 AND UAProf: MCF

A.1.2 Send Transaction

Item	Function	Reference	Status	Requirement
MMSCTR-SND-C-001	Send Transaction between Originating MMS Client and MMS Proxy-Relay	7.1	O	MMSCTR-SND-C-002 AND MMSCTR-SND-C-003
MMSCTR-SND-C-002	Originating MMS Client Sending M-Send.req to MMS Proxy-Relay	7.1.1	O	MMSCTR-PDU-C-001 AND MMSCTR-WSP-C-001

MMSCTR-SND-C-003	MMS Proxy-Relay Sending M-Send.conf to Originating MMS Client	7.1.1	O	MMSCTR-PDU-C-002 AND MMSCTR-WSP-C-002

A.1.3 Notification Transaction

Item	Function	Reference	Status	Requirement
MMSCTR-NTF-C-001	Notification Transaction between MMS Proxy-Relay and Receiving MMS Client	7.2	O	MMSCTR-NTF-C-002 AND MMSCTR-NTF-C-003
MMSCTR-NTF-C-002	MMS Proxy-Relay Sending M-Notification.ind to Receiving MMS Client	7.2.1	O	MMSCTR-PDU-C-003 AND MMSCTR-PSH-C-001 AND MMSCTR-PSH-C-002 AND MMSCTR-PSH-C-003
MMSCTR-NTF-C-003	Receiving MMS Client Sending M-NotifyResp.ind to MMS Proxy-Relay	7.2.1	O	MMSCTR-PDU-C-004 AND MMSCTR-WSP-C-001

A.1.4 Retrieve Transaction

Item	Function	Reference	Status	Requirement
MMSCTR-FTC-C-001	Retrieval Transaction between Receiving MMS Client and MMS Proxy-Relay	7.3	O	MMSCTR-FTC-C-002 AND MMSCTR-FTC-C-003 AND MMSCTR-FTC-C-004
MMSCTR-FTC-C-002	Receiving MMS Client Sending Retrieve Request to MMS Proxy-Relay	7.3.1	O	MMSCTR-WSP-C-003
MMSCTR-FTC-C-003	MMS Proxy-Relay Sending M-retrieve.conf to Receiving MMS Client	7.3.1	O	MMSCTR-PDU-C-005 AND MMSCTR-WSP-C-004
MMSCTR-FTC-C-004	Receiving MMS Client Sending M-Acknowledge.ind to MMS Proxy-Relay	7.3.1	O	MMSCTR-PDU-C-006 AND MMSCTR-WSP-C-001

A.1.5 Forward Transaction

Item	Function	Reference	Status	Requirement
MMSCTR-FWD-C-001	Forward Transaction between Forwarding MMS Client and MMS Proxy-Relay	7.4	O	MMSCTR-FWD-C-002 AND MMSCTR-FWD-C-003

MMSCTR-FWD-C-002	Forwarding MMS Client Sending M-Forward.req to MMS Proxy-Relay	7.4.1	O	MMSCTR-PDU-C-010 AND MMSCTR-WSP-C-001
MMSCTR-FWD-C-003	MMS Proxy-Relay Sending M-Forward.conf to Forwarding MMS Client	7.4.1	O	MMSCTR-PDU-C-011 AND MMSCTR-WSP-C-002

A.1.6 Delivery Report Transaction

Item	Function	Reference	Status	Requirement
MMSCTR-DRP-C-001	Delivery Report Transaction between MMS Proxy-Relay and Originating MMS Client	7.5	O	MMSCTR-DRP-C-002
MMSCTR-DRP-C-002	MMS Proxy-Relay Sending M-Delivery.ind to Originating MMS Client	7.5.1	O	MMSCTR-PDU-C-007 AND MMSCTR-PSH-C-001 AND MMSCTR-PSH-C-002 AND MMSCTR-PSH-C-003

A.1.7 Read Reports

Item	Function	Reference	Status	Requirement
MMSCTR-RRP-C-001	Ability to Generate Read Report in Receiving MMS Client	7.6.1, 7.6.3.1	O	MMSCTR-SND-C-001
MMSCTR-RRP-C-002	Ability to Receive Read Report in Originating MMS Client	7.6.1, 7.6.3.1	O	MMSCTR-NTF-C-001 AND MMSCTR-FTC-C-001
MMSCTR-RRP-C-003	Ability to handle Read Reports	7.6	O	MMSCTR-RRP-C-004 OR MMSCTR-RRP-C-005
MMSCTR-RRP-C-004	Ability to handle Read Reports in the form of a multimedia message	7.6.1, 7.6.3.1	O	MMSCTR-RRP-C-001 AND MMSCTR-RRP-C-002
MMSCTR-RRP-C-005	Ability to handle Read reports in the form of PDUs	7.6.2, 7.6.3.2	O	MMSCTR-RRP-C-006 AND MMSCTR-RRP-C-007 AND MMSCTR-RRP-C-008
MMSCTR-RRP-C-006	Ability to Generate Read Report PDU in Receiving MMS Client	7.6.2, 7.6.3.2	O	MMSCTR-PDU-C-008
MMSCTR-RRP-C-007	Ability to Send Read Report PDU from Receiving MMS Client to MMS Proxy-Relay	7.6.2, 7.6.3.2	O	MMSCTR-WSP-C-001
MMSCTR-RRP-C-008	Ability to Receive Read Report PDU in Originating MMS Client	7.6.2, 7.6.3.2	O	MMSCTR-PSH-C-001 AND MMSCTR-PSH-C-002 AND MMSCTR-PSH-C-003

A.1.8 PDU Encapsulation Dependencies

Item	Function	Reference	Status	Requirement
MMSCTR-PDU-C-001	Originating MMS Client Sending Encapsulated M-Send.req PDU to MMS Proxy-Relay	7.1.1	O	MMSE-C-016 AND MMSE-C-017 AND MMSE-C-018 AND MMSE-C-020 AND MMSE-C-024 AND MMSE-C-033 AND MMSE-C-036
MMSCTR-PDU-C-002	MMS Proxy-Relay Sending Encapsulated M-Send.conf PDU to Originating MMS Client	7.1.1	O	MMSE-C-016 AND MMSE-C-017 AND MMSE-C-018 AND MMSE-C-034 AND MMSE-C-037
MMSCTR-PDU-C-003	MMS Proxy-Relay Sending Encapsulated M-Notification.ind PDU to Receiving MMS Client	7.2.1	O	MMSE-C-042 AND MMSE-C-043 AND MMSE-C-044 AND MMSE-C-047 AND MMSE-C-048 AND MMSE-C-049 AND MMSE-C-050
MMSCTR-PDU-C-004	Receiving MMS Client Sending Encapsulated M-NotifyResp.ind PDU to MMS Proxy-Relay	7.2.1	O	MMSE-C-042 AND MMSE-C-043 AND MMSE-C-044 AND MMSE-C-051
MMSCTR-PDU-C-005	MMS Proxy-Relay Sending Encapsulated M-retrieve.conf PDU to Receiving MMS Client	7.3.1	O	MMSE-C-058 AND MMSE-C-061 AND MMSE-C-062 AND MMSE-C-063 AND MMSE-C-064
MMSCTR-PDU-C-006	Receiving MMS Client Sending Encapsulated M-Acknowledge.ind PDU to MMS Proxy-Relay	7.3.1	O	MMSE-C-083 AND MMSE-C-084 AND MMSE-C-085
MMSCTR-PDU-C-007	MMS Proxy-Relay Sending Encapsulated M-Delivery.ind PDU to Originating MMS Client	7.5.1	O	MMSE-C-083 AND MMSE-C-085 AND MMSE-C-087 AND MMSE-C-088 AND MMSE-C-089 AND MMSE-C-090
MMSCTR-PDU-C-008	Receiving MMS Client Sending Encapsulated M-Read-Rec.ind PDU to MMS Proxy-Relay	7.6.1	O	MMSE-C-060 AND MMSE-C-071 AND MMSE-RDR-C-004 AND MMSE-RDR-C-005 AND MMSE-RDR-C-006 AND MMSE-RDR-C-007 AND MMSE-RDR-C-008 AND MMSE-RDR-C-011

MMSCTR-PDU-C-009	MMS Proxy-Relay Sending Encapsulated M-Read-Orig.ind PDU to Originating MMS Client	7.6.2	O	MMSE-RDR-C-004 AND MMSE-RDR-C-005 AND MMSE-RDR-C-006 AND MMSE-RDR-C-007 AND MMSE-RDR-C-008 AND MMSE-RDR-C-010 AND MMSE-RDR-C-011
MMSCTR-PDU-C-010	MMS Client Sending Encapsulated M-Forward.req PDU to MMS Proxy-Relay	7.4.1	O	MMSE-FWD-C-002 AND MMSE-FWD-C-003 AND MMSE-FWD-C-004 AND MMSE-FWD-C-006 AND MMSE-FWD-C-015 AND MMSE-FWD-C-019
MMSCTR-PDU-C-011	MMS Proxy-Relay Sending Encapsulated M-Forward.conf PDU to MMS Client	7.4.1	O	MMSE-FWD-C-002 AND MMSE-FWD-C-003 AND MMSE-FWD-C-004 AND MMSE-FWD-C-016 AND MMSE-FWD-C-018

A.1.9 WAP PUSH Dependencies

Item	Function	Reference	Status	Requirement
MMSCTR-PSH-C-001	MMS Proxy-Relay Using WAP PUSH Operation to Send MMS PDU to MMS Client	9.1.2, 9.1.4, 9.1.5, 9.2.2, 9.2.4, 9.2.5	O	MMSCTR-PSH-C-004 OR MMSCTR-PSH-C-005 OR MMSCTR-PSH-C-008
MMSCTR-PSH-C-002	Format and Contents of Push Message	9.1.2, 9.1.4, 9.1.5, 9.2.2, 9.2.4, 9.2.5	O	MSG-GEN-C-002 AND MSG-GEN-C-003 AND MSG-GEN-C-005
MMSCTR-PSH-C-003	Push Application Addressing and Dispatching to MMS Client	9.1.2, 9.1.4, 9.1.5, 9.2.2, 9.2.4, 9.2.5	O	(OTA-WSP-C-006 AND OTA-WSP-C-007) OR OTA-HTTP-C-011
MMSCTR-PSH-C-004	Non-secure Port for Connectionless Push	9.1.2, 9.1.4, 9.1.5, 9.2.2, 9.2.4, 9.2.5	O	OTA-CL-C-001 AND OTA-CL-C-002

MMSCTR-PSH-C-005	Secure Port for Connectionless Push	7.7, 9.1.2, 9.1.4, 9.1.5, 9.2.2, 9.2.4, 9.2.5	O	OTA-CL-C-003
MMSCTR-PSH-C-008	Connection Oriented Push	7.7, 9.1.2, 9.1.4, 9.1.5, 9.2.2, 9.2.4, 9.2.5	O	OTA-CO-C-001

A.1.10 WSP/HTTP Dependencies

Item	Function	Reference	Status	Requirement
MMSCTR-WSP-C-001	MMS Client Using WSP/HTTP POST Request to Send MMS PDU to MMS Proxy-Relay	9.1.1, 9.1.2, 9.1.3, 9.1.5, 9.1.6, 9.2.1, 9.2.2, 9.2.3, 9.2.5, 9.2.6	O	WSP-C-001 OR HTTP-CLT-C-001
MMSCTR-WSP-C-002	MMS Proxy-Relay Using WSP/HTTP POST Response to Send MMS PDU to MMS Client	9.1.1, 9.1.6, 9.2.1, 9.2.6	O	WSP-C-001 OR HTTP-CLT-C-001
MMSCTR-WSP-C-003	MMS Client Using WSP/HTTP GET Request to Request MMS PDU from MMS Proxy-Relay	9.1.3, 9.2.3	O	WSP-C-001 OR HTTP-CLT-C-001
MMSCTR-WSP-C-004	MMS Proxy-Relay Using WSP/HTTP GET Response to Send MMS PDU to MMS Client	9.1.3, 9.2.3	O	WSP-C-001 OR HTTP-CLT-C-001
MMSCTR-WSP-C-006	http URI Scheme	9.3.1	M	
MMSCTR-WSP-C-007	https URI Scheme	9.3.2	M	MMSCTR-WSP-C-008 OR MMSCTR-WSP-C-009 OR MMSCTR-WSP-C-010
MMSCTR-WSP-C-008	https URI Scheme over WP-HTTP	9.3.2	O	TLS:MCF
MMSCTR-WSP-C-009	https URI Scheme over WSP	9.3.2	O	WTLS:MCF

MMSCTR-WSP-C-010	Report an Error When No TLS or WTLS Security Service is Available in the Terminal	9.3.2	O	

A.2 MMS Proxy-Relay

A.2.1 Server Level Function Groups

Item	Function	Reference	Status	Requirement
MMSCTR-SLF-S-004	Capability Negotiation between Receiving MMS Client and MMS Proxy-Relay using the UAProf Component MMS Characteristics	8, 9	O	UAProf: MSF

A.2.2 Send Transaction

Item	Function	Reference	Status	Requirement
MMSCTR-SND-S-001	Send Transaction between Originating MMS Client and MMS Proxy-Relay	7.1	M	MMSCTR-SND-S-002 AND MMSCTR-SND-S-003
MMSCTR-SND-S-002	Originating MMS Client Sending M-Send.req to MMS Proxy-Relay	7.1.1	O	MMSE-S-076
MMSCTR-SND-S-003	MMS Proxy-Relay Sending M-Send.conf to Originating MMS Client	7.1.1	O	MMSE-S-076

A.2.3 Notification Transaction

Item	Function	Reference	Status	Requirement
MMSCTR-NTF-S-001	Notification Transaction between MMS Proxy-Relay and Receiving MMS Client	7.2	M	MMSCTR-NTF-S-002 AND MMSCTR-NTF-S-003
MMSCTR-NTF-S-002	MMS Proxy-Relay Sending M-Notification.ind to Receiving MMS Client	7.2.1	O	MMSE-S-077 AND MMSCTR-PSH-S-002
MMSCTR-NTF-S-003	Receiving MMS Client Sending M-NotifyResp.ind to MMS Proxy-Relay	7.2.1	O	MMSE-S-077

A.2.4 Retrieve Transaction

Item	Function	Reference	Status	Requirement
MMSCTR-FTC-S-001	Retrieval Transaction between Receiving MMS Client and MMS Proxy-Relay	7.3	M	MMSCTR-FTC-S-002 AND MMSCTR-FTC-S-003 AND MMSCTR-FTC-S-004
MMSCTR-FTC-S-002	Receiving MMS Client Sending Retrieve Request to MMS Proxy-Relay	7.3.1	O	
MMSCTR-FTC-S-003	MMS Proxy-Relay Sending M-retrieve.conf to Receiving MMS Client	7.3.1	O	MMSE-S-078
MMSCTR-FTC-S-004	Receiving MMS Client Sending M-Acknowledge.ind to MMS Proxy-Relay	7.3.1	O	MMSE-S-078

A.2.5 Forward Transaction

Item	Function	Reference	Status	Requirement
MMSCTR-FWD-S-001	Forward Transaction between Forwarding MMS Client and MMS Proxy-Relay	7.4	O	MMSCTR-FWD-S-002 AND MMSCTR-FWD-S-003
MMSCTR-FWD-S-002	Forwarding MMS Client Sending M-Forward.req to MMS Proxy-Relay	7.4.1	O	MMSE-S-097
MMSCTR-FWD-S-003	MMS Proxy-Relay Sending M-Forward.conf to Forwarding MMS Client	7.4.1	O	MMSE-S-097

A.2.6 Delivery Report Transaction

Item	Function	Reference	Status	Requirement
MMSCTR-DRP-S-001	Delivery Report Transaction between MMS Proxy-Relay and Originating MMS Client	7.5	M	MMSCTR-DRP-S-002
MMSCTR-DRP-S-002	MMS Proxy-Relay Sending M-Delivery.ind to Originating MMS Client	7.5.1	O	MMSE-S-079 AND MMSCTR-PSH-S-002

A.2.7 WAP PUSH Dependencies

Item	Function	Reference	Status	Requirement
MMSCTR-PSH-S-002	Format and Contents of Push Message	9.1.2, 9.1.4, 9.1.5, 9.2.2, 9.2.4, 9.2.5	O	MSG-GEN-S-002 AND MSG-GEN-S-003 AND MSG-GEN-S-005

A.2.8 Read Reports

Item	Function	Reference	Status	Requirement
MMSCTR-RRP-S-001	Read Report PDU transaction	7.6	M	MMSCTR-RRP-S-002 AND MMSCTR-RRP-S-003
MMSCTR-RRP-S-002	Ability to handle Read Report PDU	7.6.2, 7.6.3.2	O	MMSE-S-091 AND MMSE-S-092
MMSCTR-RRP-S-003	Ability to send M-Read-Orig.ind PDU from MMS Proxy-Relay to Originating MMS Client	7.6.2, 7.6.3.2	O	MMSCTR-PSH-S-002
MMSCTR-RRP-S-004	Ability to transform an M-Read-Orig.ind PDU into a multimedia message.	7.6.2, 7.6.3.2	O	

A.2.9 WSP/HTTP Dependencies

Item	Function	Reference	Status	Requirement
MMSCTR-WSP-S-006	http URI Scheme	9.3.1	M	
MMSCTR-WSP-S-007	https URI Scheme	9.3.2	O	MMSCTR-WSP-S-008
MMSCTR-WSP-S-008	https URI Scheme over HTTP	9.3.2	O	TLS:MSF

Appendix B. UAProf Schema for MMS Characteristics (Normative)

```
<?xml version="1.0"?>
       <rdf:RDF xmlns:rdf = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
       xmlns:rdfs = "http://www.w3.org/2000/01/rdf-schema#"
       xmlns:prf="http://www.wapforum.org/profiles/MMS/ccppschema-20010111#">
       <rdf:Description ID="Component">
              <rdf:type resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
              <rdfs:subClassOf rdf:resource="http://www.w3.org/2000/01/rdf-
                     schema#Resource"/>
                     <rdfs:label>Component</rdfs:label>
                     <rdfs:comment>
                            A Component within the CC/PP Schema is a class of related
                            properties that describe the capabilities and preferences
                            information.
                     </rdfs:comment>
       </rdf:Description>


<!-- ************************************************************** -->
<!-- ***** Properties shared among the components***** -->

<rdf:Description ID="component">
<rdf:type resource="http://www.w3.org/2000/01/rdf-schema#Property"/>
<rdfs:label>component</rdfs:label>
<rdfs:comment>
       The component attribute links the various components to
       the root node(profile).
</rdfs:comment>
</rdf:Description>



<!-- ************************************************************** -->

<!-- ***** Component Definitions ***** -->



<rdf:Description ID="MmsCharacteristics">

       <rdf:type resource="http://www.w3.org/2000/01/rdf-schema#Class"/>

       <rdfs:subClassOf rdf:resource="#Component"/>

       <rdfs:label>Component: MmsCharacteristics</rdfs:label>

       <rdfs:comment>

              The MmsCharacteristics component contains properties of the device's

              Multimedia messaging capabilities, such as maximum message size, maximum

              image resolution, etc.

       </rdfs:comment>

</rdf:Description>


<!-- ************************************************************** -->
<!-- ***** Component: MmsCharacteristics ***** -->
<!-- ************************************************************** -->
<!-- ***** Attributes for component:   MmsCharacteristics ***** -->

<rdf:Description ID="MmsMaxMessageSize">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Property"/>
   <rdfs:domain rdf:resource="#MmsCharacteristics"/>
          <rdfs:comment>
                Description: Maximum size of an MMS message in bytes.
```

```
                Type:                           Number
                Resolution:   Locked
                Examples:     2048
        </rdfs:comment>
</rdf:Description>


<rdf:Description ID="MmsMaxImageResolution">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Property"/>
   <rdfs:domain rdf:resource="#MmsCharacteristics"/>
        <rdfs:comment>
             Description:  The maximum size of an image in units of pixels
                                                    (horizontal x vertical).
                Type:                           Literal
                Resolution:   Locked
                Examples:     80x60
        </rdfs:comment>
</rdf:Description>



<rdf:Description ID="MmsCcppAccept">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Property"/>
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Bag"/>
   <rdfs:domain rdf:resource="#MmsCharacteristics"/>
        <rdfs:comment>
                        Description: Property value is a list of supported content types
                                                    where each item in the list is a
content type name
                                                    registered as a MIME type
                Type:                           Literal bag
                Resolution:   Locked
                Examples:     “image/jpeg”, “audio/wav”, “video/mpeg-4”
        </rdfs:comment>
</rdf:Description>


<rdf:Description ID="MmsCcppAcceptCharSet">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Property"/>
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Bag"/>
   <rdfs:domain rdf:resource="#MmsCharacteristics"/>
        <rdfs:comment>
                        Description: List of character sets that the MMS Client supports.
                                                    Property value is a list of
character sets, where
                                                    each item in the list is a
character set name registered
                                                    with IANA
                Type:                           Literal bag
                Resolution:   Locked
                Examples:     “US-ASCII”, “ISO-8859-1”
        </rdfs:comment>
</rdf:Description>

<rdf:Description ID="MmsCcppAcceptLanguage">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Property"/>
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Bag"/>
   <rdfs:domain rdf:resource="#MmsCharacteristics"/>
        <rdfs:comment>
                        Description: List of preferred languages. The first item in the
                                                    list should be considered the
user’s first choice.
                                                    Property value is a list of
natural languages, where
                                                    each item in the list is the name
of a language as
                                                    defined by RFC 1766.
                Type:                           Literal bag
                Resolution:   Locked
                Examples:     “en”, “fr”
        </rdfs:comment>
```

```
</rdf:Description>

<rdf:Description ID="MmsCcppAcceptEncoding">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Property"/>
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Bag"/>
   <rdfs:domain rdf:resource="#MmsCharacteristics"/>
          <rdfs:comment>
                        Description: List of transfer encodings that the MMS Client supports.
                                                              Property value is a list of
transfer encodings, where
                                                              each item in the list is a
transfer encoding name as
                                                              specified by RFC 2045 and
registered with IANA.
                Type:                         Literal bag
                Resolution:  Locked
                Examples:    “base64”, “quoted-printable"
          </rdfs:comment>
</rdf:Description>

<rdf:Description ID="MmsVersion">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Property"/>
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Bag"/>
   <rdfs:domain rdf:resource="#MmsCharacteristics"/>
          <rdfs:comment>
                        Description: The MMS versions supported by the MMS Client conveyed
                                                              as
majorVersionNumber.minorVersionNumber.
                Type:                         Literal bag
                Resolution:  Locked
                Examples:    "2.0", "1.3"
          </rdfs:comment>
</rdf:Description>

<rdf:Description ID="MmsCcppStreamingCapable">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Property"/>
   <rdfs:domain rdf:resource="#MmsCharacteristics"/>
          <rdfs:comment>
                        Description: Indicates whether the MMS Client is capable of invoking
                              streaming.
                Type:                         Boolean
                Resolution:  Locked
                Examples:    "Yes", "No"
          </rdfs:comment>
</rdf:Description>

</rdf:RDF>
```

Appendix C. Relationship to 3GPP MMS Specifications (Informative)

The client transactions defined in this document aim at fulfilling the requirements stated in the technical specifications of the 3rd Generation Partnership Project. These include the service aspects of MMS described in [TS22140] and the functional description contained in [TS23140].

In order to provide for an unambiguous relationship between WAP MMS PDUs and 3GPP MMS messages this appendix contains a mapping between these PDUs and messages.

WAP PDUs are defined for the interface between the MMS Proxy-Relay and the MMS Client. This interface corresponds to the interface between the MMS Relay/Server and the MMS User Agent, called "MM1"in the 3GPP specifications.

The following sections cover the different transactions defined in this specification.

NOTE: The distinction between immediate and deferred retrieval is only done in the WAP MMS specifications. Therefore the relationship between WAP MMS PDUs and 3GPP MMS messages are slightly different for the two retrieval cases.

a) MMS Client Sending Message to MMS Proxy-Relay

The following table figures out the relationship between WAP MMS PDUs and the corresponding 3GPP MMS messages when the MMS Client sends a multimedia message to the MMS Proxy Relay.

WAP MMS PDU	3GPP MMS Message
M-Send.req	MM1_submit.REQ
M-Send.conf	MM1_submit.RES

Table 3: Relationship between WAP MMS PDUs and 3GPP MMS messages when the MMS Client sends a multimedia message to the MMS Proxy Relay.

b) MMS Proxy-Relay Sending Notification to MMS Client in case of Deferred Retrieval

The following table figures out the relationship between WAP MMS PDUs and the corresponding 3GPP MMS messages when the MMS Proxy-Relay sends a notification to the MMS Client in case of deferred retrieval.

WAP MMS PDU	3GPP MMS Message
M-Notification.ind	MM1_notification.REQ
M-NotifyResp.ind	MM1_notification.RES

Table 4: Relationship between WAP MMS PDUs and 3GPP MMS messages when the MMS Proxy-Relay sends a notification to the MMS Client in case of deferred retrieval.

c) MMS Proxy-Relay Sending Notification to MMS Client in case of Immediate Retrieval

The following table figures out the relationship between WAP MMS PDUs and the corresponding 3GPP MMS messages when the MMS Proxy-Relay sends a notification to the MMS Client in case of immediate retrieval.

WAP MMS PDU	3GPP MMS Message
M-Notification.ind	MM1_notification.REQ
M-NotifyResp.ind	MM1_acknowledgement.REQ

Table 5: Relationship between WAP MMS PDUs and 3GPP MMS messages when the MMS Proxy-Relay sends a notification to the MMS Client in case of immediate retrieval.

d) MMS Client Retrieving Message from MMS Proxy-Relay

The following table figures out the relationship between WAP MMS PDUs and the corresponding 3GPP MMS messages when the MMS Client retrieves the multimedia message from the MMS Proxy-Relay.

WAP MMS PDU	3GPP MMS Message
WSP/HTTP-Get.req	MM1_retrieve.REQ
M-Retrieve.conf	MM1_retrieve.RES
M-Acknowledge.ind	MM1_acknowledgement.REQ

Table 6: Relationship between WAP MMS PDUs and 3GPP MMS messages when the MMS Client retrieves the message from the MMS Proxy-Relay.

NOTE: In case of immediate retrieval the PDU M-Acknowledge.ind is omitted from this transaction. Its task is fulfilled by the M-NotifyResp.ind contained in the transaction of sending a notification from the MMS Proxy-Relay to an MMS Client described in the section above.

e) Message Forwarding

The following table figures out the relationship between WAP MMS PDUs and the corresponding 3GPP MMS messages in the case of message forwarding without prior retrieval of the message.

WAP MMS PDU	3GPP MMS Message
M-Forward.req	MM1_forward.REQ
M-Forward.conf	MM1_forward.RES

Table 7: Relationship between WAP MMS PDUs and 3GPP MMS messages in case of message forwarding.

f) MMS Proxy-Relay Sending Delivery Report to MMS Client

The following table figures out the relationship between WAP MMS PDU and the corresponding 3GPP MMS message when the MMS Proxy-Relay sends a delivery report to the MMS Client.

WAP MMS PDU	3GPP MMS Message
M-Delivery.ind	MM1_delivery_report.REQ

Table 8: Relationship between WAP MMS PDU and 3GPP MMS message when the MMS Proxy-Relay sends a delivery report to the MMS Client.

g) Recipient MMS Client Sending Read Report to Originator MMS Client

The following table figures out the relationship between WAP MMS PDUs and the corresponding 3GPP MMS messages when the recipient MMS Client sends a read report as a regular MM to the originator MMS Client via one or more MMS Proxy-Relay(s).

WAP MMS PDU	3GPP MMS Message
new MMS-Message: M-Send.req	MM1_read_reply_receipient.REQ
M-Notification.ind + M-Retrieve.conf	MM1_read_reply_originator.REQ

Table 9: Relationship between WAP MMS PDUs and 3GPP MMS messages when the recipient MMS Client sends a read report as a regular MM to the originator MMS Client via one or more MMS Proxy-Relay(s).

The following table figures out the relationship between WAP MMS PDUs and the corresponding 3GPP MMS messages when the recipient MMS Client sends a read report as a PDU to the originator MMS Client via one or more MMS Proxy-Relay(s).

WAP MMS PDU	3GPP MMS Message
M-ReadRec.ind	MM1_read_reply_receipient.REQ
M-ReadOrig.ind	MM1_read_reply_originator.REQ

Table 10: Relationship between WAP MMS PDUs and 3GPP MMS messages when the recipient MMS Client sends a read report as a PDU to the originator MMS Client via one or more MMS Proxy-Relay(s)

Appendix D. Change History (Informative)

D.1 Approved Version History

Reference	Date	Description
WAP-206-MMSCTR-20010612-a	12 Jun 2001	
WAP-206-MMSCTR-20020115-a	15 Jan 2002	
OMA-WAP-CTR-SCD_100-20020909-a	09 Sep 2002	
OMA-WAP-MMS-CTR-V1_1	15 Jul 2004	Status changed to Approved by TP TP ref # OMA-TP-2004-0227-MMS-V1_1-for-final-approval

D.2 Draft/Candidate Version 1.1 History

Document Identifier	Date	Sections	Description
Candidate Version OMA-WAP-MMS-CTR-V1_1	31 Oct 2002	n/a	