# MMS Message Template Specification

Candidate Version 1.3 – 28 Jan 2008

## Open Mobile Alliance

OMA-TS-MMS-TEMP-V1_3-20080128-C

# Contents



# Tables

# 1. Scope

The Open Mobile Alliance (OMA) specifications are the result of an ongoing effort to define mechanisms for enabling industry-wide interoperable applications and services for deployment over wireless networks.

To continue the enhancement and evolution of the Multimedia Messaging Service OMA is developing a new set of specifications that will define the MMS v1.3 release. This release will expand and build upon the existing MMS v1.2 specifications.

The functionality of the MMS 1.3 release, while emphasizing backward compatibility and interoperability, is enhanced to improve the user experience with richer content capability, especially with regard to imaging and presentation features. The concept of Template is introduced in MMS 1.3 as an enabling mechanism to facilitate implementation of applications utilizing many of the new features in the release.

This document describes the MMS Message Template profile specification and its conformance. The MMS Message Template Profile specifies a new frame work to enable template functionality for MMS. This new functionality provides a simple method for end users to compose rich MM and for service providers to widely distribute new services by leveraging the interoperability guaranteed by MMS 1.3.

# 2. References

## 2.1 Normative References


**[MMSCONF]** "OMA MMS Conformance Document", OMA-TS-MMS-CONF-V1_3, Open Mobile Alliance™. URL: http://www.openmobilealliance.org

**[MMSCTR]** "MMS Client Transactions", OMA-TS-MMS-CTR-V1_3, Open Mobile Alliance™. URL: http://www.openmobilealliance.org

**[MMSENCAPS]** "MMS Encapsulation Protocol", OMA-TS-MMS-ENC-V1_3, Open Mobile Alliance™. URL: http://www.openmobilealliance.org

**[MMSMTDDTD]** "DTD for the MMS Message Template Definition", OMA-SUP-DTD_mms_mtd-V1_3, Open Mobile Alliance™. URL: http://www.openmobilealliance.org

**[MMSMTDSCHEMA]** "XML Schema for the MMS Message Template Definition", OMA-SUP-XSD_mms_mtd-V1_3, Open Mobile Alliance™. URL: http://www.openmobilealliance.org

**[OMADRM]** "Digital Rights Management", version 1.0, OMA-DRM-V1_0, Open Mobile Alliance ™. URL; http://www.openmobilealliance.org

**[RFC2387]** "The MIME Multipart/Related Content-type", IETF, RFC 2387, URL: http://www.ietf.org/rfc/rfc2387.txt

**[RFC2396]** "Uniform Resource Identifiers (URI): Generic Syntax", IETF, RFC 2396, URL: http://www.ietf.org/rfc/rfc2396.txt

**[RFC2557]** "MIME Encapsulation of Aggregate Documents, such as HTML (MHTML)", IETF, RFC 2557, URL: http://www.ietf.org/rfc/rfc2557.txt

**[RFC3023]** "XML Media Types", IETF, RFC3023, URL: http://www.ietf.org/rfc/rfc3023.txt

**[SMIL]** "Synchronized Multimedia Integration Language (SMIL 2.0)", W3C Recommendation 07 August 2001, URL: http://www.w3.org/TR/smil20/

**[TS23140]** "Multimedia Messaging Service: Functional description; Stage 2", 3rd Generation Partnership Project, TS 23.140 Release 5. URL: http://www.3gpp.org/ftp/Specs/

**[TS26140]** "Multimedia Messaging Service; Media formats and codecs", 3rd Generation Partnership Project, TS 26.140 Release 5. URL: http://www.3gpp.org/ftp/Specs/

**[Unicode]** "The Unicode Standard Version 3.0", The Unicode Consortium, Addison-Wesley, Reading (MA), January 2000. ISBN 0-201-61633-5.

**[VOBJECT]** "vObject Minimum Interoperability Profile", version 1.0, OMA-vObject-V1_0, Open Mobile Alliance ™. URL: http://www.openmobilealliance.org

**[XML]** "Extensible Markup Language (XML) 1.0 (Second Edition)", W3C Recommendation, Tim Bray et al., 6 October 2000. URL: http://www.w3.org/TR/2000/REC-xml-20001006


## 2.2 Informative References


**[RFC2046]** "MIME Part Two: Media Types", IETF, RFC 2046, URL: http://www.ietf.org/rfc/rfc2046.txt

# 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

| | |
|---|---|
| **MMS Message Template** | A set of template materials that contains MTD, MMS presentation part, and media objects. |
| **MTD** | An XML document that describes the components that make up an MMS Message Template as defined in this specification. |
| **MMS client with MMS Message Template functionality** | A client needed to use the MTD to build an MM that is compliant to the Core MM Content Domain as defined in [MMSCONF]. The MMS client with MMS Message Template functionality is referred to as MMS Client in this document |

## 3.3 Abbreviations

| | |
|---|---|
| **DRM** | Digital Rights Management |
| **HTTP** | HyperText Transfer Protocol |
| **MIME** | Multipurpose Internet Mail Extension |
| **MM** | Multimedia Message |
| **MMS** | Multimedia Messaging Service |
| **MMSIOP** | MMS Interoperability between MMS handsets and MMS Servers |
| **MTD** | Message Template Definition |
| **OMA** | Open Mobile Alliance |
| **RFC** | Request For Comments |
| **SMIL** | Synchronized Multimedia Integration Language |
| **URI** | Uniform Resource Identifier |
| **URL** | Uniform Resource Locator |
| **UTF-8** | Unicode Transformation Format |
| **W3C** | World Wide Web Consortium |
| **WAP** | Wireless Application Protocol |
| **XML** | Extensible Markup Language |

# 4. Introduction

The concept of MMS Message Template is introduced in MMS as an enabling mechanism to facilitate implementation of applications utilizing many of the new features in the release. This new template functionality provides a simple method for end users to compose rich MMs and for service providers to widely distribute new services by leveraging the interoperability guaranteed by MMS.

MMS Message Template is defined as a structure, activated in a user device but created by a "service", that serves as a model or pattern to allow the user to easily aggregate in a coherent and pleasing manner elements of different MM content classes for transmission to another user or service in an MM. Examples of applications using MMS Message Templates are MMS postcards, electronic greeting cards and multimedia stationery.

The Multimedia Messaging Service (MMS) architecture [MMSCONF] uses the MMS SMIL Language Profile to describe MMs that can be sent to and from mobile terminals. As mobile terminals become more capable, the MMS SMIL Language Profile has become more robust in order to deliver increasingly rich MMs.

Composing a message that uses the complex features of SMIL is not a simple cognitive task. As a result, this complexity may limit the usefulness of these features in the mass market.

MMS Message Templates provide a means for simplifying the composition of feature-rich MMs.

The MMS message template has functionality to handle MMS SMIL documents as an interface. It also has functionality to invoke device applications such as camera, sound recorder, text editor, and address book. By gathering together various media objects derived from user inputs and device applications, the MMS Message Template client composes an MM that is compliant to the Core MM Content Domain defined in [MMSCONF].

Service providers can create and then provide user terminals with MMS Message Templates that describe arbitrarily complex MMs. Subscribers' clients interact with the MMS Message Template and the user in the process of building an MMS SMIL Language Profile compliant MM that conforms to the OMA MMS Conformance specification [MMSCONF].

The support of the MMS Template specification is optional for MMS Clients compliant with MMS V1_3 Enabler Release.

# 5. MMS Message Template Specification

This section defines the "MMS Message Template Specification", a language profile that supports MMS requirements.

## 5.1 Reference Processing Model (Informative)

Section 5.1 assumes the implementation includes a “Wizard part” as defined in 5.2.1.5. However, the Wizard part is an optional functionality of the overall MMS Client, and this should be taken into account when considering the discussion in this section.

Although Wizard is an optional functionality in the MMS Template Specification, it is assumed in the specification that some form of interactivity with the user is present to support the replacement of media objects in the MM.

The MMS Message Template profile reference-processing model consists of eight (8) entities:

1. MMS Client
2. MTD
3. MM
4. Resulting MM
5. MMS presentation part
6. Fixed media objects
7. Replaceable media objects
8. User-specified media objects

The MMS Client is an MMS client that implements all or part of the MMS Message Template solution.

The MTD describes the meta information of the template and defines a wizard to compose a message. This MTD is typically created by a service and delivered to a terminal as an MM that is processed by the MMS Client. MTD may also be pre-stored in a terminal. The MTD may be protected by DRM AS DEFINED IN [OMADRM].

The MM is a multimedia message that is compliant to the Core MM Content Domain defined in [MMSCONF]. An MMS Message Template is created by integrating an MTD into an MM. The MMS Message Template, in addition to an MTD, may contain an MMS presentation part, some fixed media objects, and replaceable media objects. The MM encode type MAY be multipart/related or multipart/mixed as defined in [MMSCONF]. The encode type of the Resulting MM depends on the original MM.

The Resulting MM is a composed MM that can be sent to another MMS client. The media objects in the Resulting MM may be protected by DRM AS DEFINED IN [OMADRM].

MMS presentation is a presentation of the MM defined in [MMSCONF] that will be used for the Resulting MM. In MMS1.3, it SHALL be an MMS SMIL document. The MMS presentation + Fixed media objects + User-specified media objects is the Resulting MM from user interaction with MMS Message Template.

Fixed media objects are those media objects in an MM that cannot be replaced by the end user. In often cases, these media objects are protected by DRM AS DEFINED IN [OMADRM].

Replaceable media objects are those media objects in an MM that can be replaced by the end user.

User-specified media objects are those media objects chosen by the user for use in the Resulting MM.

The reference-processing model has 3 phases:

Phase 1: Identify needed resources

Phase 2: Acquire replaceable media objects

Phase 3: Create Resulting MM

The MMS Client controls the execution of these three phases and any interaction between the mobile terminal and the end user.

### 5.1.1 Phase 1 – Identify Needed Resources

In Phase 1, MMS Client selects an appropriate MTD. How this selection occurs is beyond the scope of this reference-processing model. All needed resources SHALL be located in the terminal before template usage.

The MTD identifies:

- Zero or one MMS presentation part
- One or more wizard steps
- The first wizard step
- Zero or more fixed media objects
- Zero or more replaceable media objects

### 5.1.2 Phase 2 – Acquire Replaceable Media Objects

In Phase 2, MMS Client displays the first (and perhaps only) step of the wizard.

Each step defines an interaction with the end user. It may invoke various device applications such as camera that allows the user to input various media objects.

It is likely that an MMS Message Template will contain both replaceable media objects as well as fixed media objects. Replaceable media objects are "default" or "placeholder" assets that the user will likely change or customize. Fixed media objects are those assets that the end user cannot modify. For example, a "birthday card" service might have a text and image object as replaceable, but the background music is fixed.

During Phase 2, the end user interacts with the each wizard step, filling in text fields, selecting and acquiring images, etc.

Upon completion of the first step, the results are locally stored as User-specified media objects. Subsequent steps are then displayed and end user interaction continues until composition is complete.

The MTD indicates the criteria for completing composition.

### 5.1.3 Phase 3 – Create Resulting MM

In Phase 3, MMS Client uses the MMS presentation part to assemble the fixed and user-specified media objects into a Resulting MM.

## 5.2 MMS Message Template Specification (Normative)

This document defines two normative entities:

1. MTD
2. MMS Client

The MTD is a structure that describes the components that make up an MMS Message Template as defined in this specification. The MTD is a new MIME content type designed to be delivered in an MM as part of a multipart/mixed and multipart/related.

The MMS Client is needed to use the MTD to build an MM that is compliant to the Core MM Content Domain as defined in [MMSCONF].

### 5.2.1 MTD

The MTD is an XML document that identifies:

- MMS Message Template meta information
- Predefined MMS header values
- Wizard to compose a message

*Note: Wizard is an optional functionality to this specification. However, if Wizard is implemented, the functionality described below is normative and indicates a minimum set of functions necessary to satisfy the specification.*

The following example illustrates an MTD:

**postcard-sample.mtd**

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mmstemplate PUBLIC "-//OMA//DTD MMS MTD 1.3//EN"
"http://www.openmobilealliance.org/tech/DTD/mms_mtd-v1_3.dtd">
<mmstemplate xmlns="urn:oma:xml:mms:mtd:1.3">
   <head>
      <title>Postcard Example</title>
      <description>MTD sample code for Postcard service</description>
      <date>2004-10-12</date>
      <version>1.02</version>
      <author>Hirotaka UEDA</author>
   </head>
   <body>
      <message>
         <to-header editable="false">18175551212</to-header>
         <subject-header>POSTCARD</subject-header>
      </message>
      <wizard>
         <step guide="Please input the recipient address" app="text-input" target-name="address.txt"
target-type="text/plain" required=”true”/>
         <step guide="Please take a photo" app="still-camera" param="resolution=1024x768" target-
name="image.jpg" target-type="image/jpeg"/>
         <step guide="Please input your message" app="rich-text-input" target-name="message.html" target-
type="text/html"/>
      </wizard>
   </body>
</mmstemplate>
```

As this example illustrates, the MTD consists of two main parts: the head and the body, with the body being composed of a message element and a wizard element. The head part provides information about the template itself rather than the content of the template; it essentially provides meta information about the template The message part within the body indicates any predefined header values. The wizard part defines a wizard mechanism to help and guide the user in composing a message for the particular application at hand. For example, the wizard element guides the user for text, applications, such as camera, etc.

The following element and attribute definitions refer to requirements on the MMS Message Template language, rather than the requirements of a client processing an MTD. Just because an element or attribute is required does not mean that a conformant client must process that element or attribute. See Section 5.2.2 for information about client conformance.

The MTD SHALL conform to the constraints expressed in the XML Schema found in[MMSMTDSCHEMA].

The MIME type for MTD is "application/vnd.omammsg-mtd+xml".

MTD SHALL have the extension ".mtd" (all lowercase).

The following are the MTD namespace, public identifier and system identifier:

MTD Namespace:

urn:oma:xml:mms:mtd:1.3

Public Identifier for MTD 1.3:

PUBLIC "-//OMA//DTD MMS_MTD 1.3//EN"

System Identifier for MTD 1.3: http://www.openmobilealliance.org/tech/DTD/mms_mtd-v1_3.dtd

The name of the default namespace on the root element SHALL be the MTD namespace name.

The MTD SHALL have a DOCTYPE declaration in the document prior to the root element, with the public identifier. The public identifier included in the DOCTYPE declaration SHALL reference the DTD found in [MMSMTDDTD] using its Formal Public Identifier. The system identifier MAY be modified appropriately.

The following is an example document type declaration for an MTD document:

```
<!DOCTYPE mmstemplate PUBLIC "-//OMA//DTD MMS MTD 1.3//EN"
"http://www.openmobilealliance.org/tech/DTD/mms_mtd-v1_3.dtd ">
```

Note: When using [MMSMTDSCHEMA], the document type declaration for an MTD document is not necessary. The following is an example of XML schema binding for an MTD document.

```
<mmstemplate xmlns="urn:oma:xml:mms:mtd:1.3"
xmlns:xsi=”http://www.w3.org/2001/XMLSchema-instance”
xsi:schemaLocation=”http://www.openmobilealliance.org/tech/profiles/mms_mtd-v1_3.xsd”>
```

#### 5.2.1.1 The Root

The **mmstemplate** element is the root element that indicates that this document is an MTD.

The **mmstemplate** element SHALL contain two child elements: **head** and **body**. The **head** element defines the head part of the document. The **body** element defines the body of the document.

#### 5.2.1.2 The Head Part

The head part of the MTD supports the following elements: **head**, **title**, **description**, **date**, **version**, **author**, and **meta.**

##### The head element

The **head** element is used to provide descriptive information about the template itself, rather than information about the content or nature of the template. The **head** element is a child of the **mmstemplate** element.

The **head** element SHALL contain the **title** element.

The **head** element MAY contain a single **description** element.

The **head** element MAY contain a single **date** element.

The **head** element MAY contain a single **version** element.

The **head** element MAY contain a single **author** element.

The **head** element MAY contain zero or more **meta** elements.

The **head** element MAY have a **xml:lang** attribute value. The **xml:lang** attribute specifies the language of the enclosed elements.

### The title element

The **title** element is used to provide a human-readable name for the template. The intention is for this name to be displayed in menus that allow end users to select a template. The **title** element is a child of the **head** element.

The **title** element SHALL contain an **xs:string**. Empty titles are not allowed.

The **title** element MAY have a **xml:lang** attribute value. The **xml:lang** attribute specifies the language of the enclosed string.

### The description element

The **description** element is used to provide a human-readable explanation of the template's purpose (i.e., the description is intended to be displayable to the end-user). The **description** element is a child of the **head** element.

When present, the **description** element SHALL contain an **xs:string**. Empty descriptions are not allowed.

The **description** element MAY have a **xml:lang** attribute value. The **xml:lang** attribute specifies the language of the enclosed string.

### The date element

The **date** element is used to provide machine-readable identification of the template's creation date. The **date** element is a child of the **head** element.

When present, the **date** element SHALL contain an **xs:date**. Empty date is not allowed.

### The version element

The **version** element is used to provide machine-readable identification of the template's version (i.e., the version of the template content, not the version of the template definition language). The **version** element is a child of the **head** element.

When present, the **version** element SHALL contain an **xs:float**. Empty versions are not allowed.

### The author element

The **author** element is used to provide a human-readable author's information such as author name, e-mail address, and so on. The **author** element is a child of the **head** element.

When present, the **author** element SHALL contain an **xs:string**. Empty authors are not allowed.

The **author** element MAY have a **xml:lang** attribute value. The **xml:lang** attribute specifies the language of the enclosed string.

### The meta element

The **meta** element is used to provide various meta information about the template. The **meta** element is a child of the **head** element. The **meta** element is an empty element. Each **meta** element specifies a single property/value pair in the name and content attributes, respectively.

The **meta** element SHALL have both **content** attribute and **name** attribute. The **content** attribute specifies a property's value that is **xs:string**. The **name** attribute identifies a property name that is **xs:string**. The list of properties for the name attribute is open-ended.

For security reasons, the **meta** element SHALL NOT contain executable script.

*Note1: The **meta** element is a generic mechanism for specifying meta data. MMS Clients are not required to support meta data mechanisms. For those that choose to support meta data, this specification does not define how meta data should be interpreted.*

#### 5.2.1.3 The Body

The body of the MTD consists of 2 parts: message and wizard part that are described below.

##### The body element

The **body** element identifies the body of the document. The **body** element is a child of **mmstemplate** element.

The **body** element MAY contain a single **message** element.

The **body** element MAY contain a single **wizard** element.

The **message** element defines the message part of the document. The **wizard** element defines the wizard part of the document.

The **body** element MAY have a **xml:lang** attribute value. The **xml:lang** attribute specifies the language of the enclosed elements.

#### 5.2.1.4 The Message Part

The message part of the MTD supports the following elements: **message**, **to-header**, **cc-header**, and **subject-header**.

##### The message element

The **message** element specifies various settings of the Resulting MM. The **message** element is a child of **body** element.

The **message** element MAY contain zero or more **to-header** elements.

The **message** element MAY contain zero or more **cc-header** elements.

The **message** element MAY contain a single **subject-header** element.

The **message** element MAY have a **xml:lang** attribute value. The **xml:lang** attribute specifies the language of the enclosed elements.

##### The to-header element

The **to-header** element is used to predefine the "To:" header value and indicate whether this value can be modified by the end-user. The **to-header** element is a child of the **message** element.

When present, the **to-header** SHALL contain a **xs:string** element content.

When present, the **to-header** element MAY have an **editable** attribute value. The **editable** attribute indicates whether this header is considered editable.

When present, the **editable** attribute SHALL contain a **xs:boolean** value. A "true" value indicates that this header value is considered editable. A "false" value indicates that this header value is not considered editable. If the attribute is not specified, the effect is as if a value of "false" were specified.

##### The cc-header element

The **cc-header** element is used to predefine the "Cc:" header value and indicate whether this value can be modified. The **cc-header** element is a child of the **message** element.

When present, the **cc-header** SHALL contain a **xs:string** element content.

When present, the **cc-header** element MAY have an **editable** attribute value. The **editable** attribute indicates whether this header is considered editable.

When present, the **editable** attribute SHALL contain a **xs:boolean** value. A "true" value indicates that this header value is considered editable. A "false" value indicates that this header value is not considered editable. If the attribute is not specified, the effect is as if a value of "false" were specified.

### The subject-header element

The **subject-header** element is used to predefine the "Subject:" header value and indicate whether this value can be modified. The **subject-header** element is a child of the **message** element.

When present, the **subject-header** element SHALL contain a **xs:string** element content.

When present, the **subject-header** element MAY have an **editable** attribute value. The **editable** attribute indicates whether this header is considered editable.

When present, the **editable** attribute SHALL contain a **xs:boolean** value. A "true" value indicates that this header value is considered editable. A "false" value indicates that this header value is not considered editable. If the attribute is not specified, the effect is as if a value of "false" were specified.

The **subject-header** element MAY have a **xml:lang** attribute value. The **xml:lang** attribute specifies the language of the enclosed string.

## 5.2.1.5 The Wizard Part

Note: Wizard is an optional functionality to this specification. However, if Wizard is implemented, the functionality described in this section is normative and indicates a minimum set of functions necessary to satisfy the specification.

The wizard part of the MTD supports the following elements: **wizard** and **step.**

### The wizard element

The **wizard** element is used to define a wizard to compose a message. The **wizard** element is a child of the **body** element.

The **wizard** element MAY contain one or more **step** elements.

The **wizard** element MAY have a **xml:lang** attribute value. The **xml:lang** attribute specifies the language of the enclosed elements.

### The step element

The **step** element is used to define each step of the wizard. The **step** element is a child of the **wizard** element.

The **step** element MAY have a **guide** attribute value. MMS Client MAY use this description to guide the user as to what to do next. For example, a dialog which says "Please take a photo" is displayed prior to activation of a camera application.

The **step** element SHALL have an **app** attribute value. The **app** attribute identifies a candidate application that should be used to interact with a user. The **app** attribute SHALL contain **xs:string** which is one of following attribute values.

- still-camera: MMS Client invokes still camera application. The application is expected to return Still Image or Bitmap as defined in [MMSCONF] and the content type is specified by **target-type** attribute.
- video-camera: MMS Client invokes video camera application. The application is expected to return Video as defined in [MMSCONF] and the content type is specified by **target-type** attribute.
- text-input: MMS Client invokes text editor. The application is expected to return Plain Text as defined in [MMSCONF] and the content type is specified by **target-type** attribute.
- rich-text-input: MMS Client invokes rich text editor. The application is expected to return Rich Text as defined in [MMSCONF] and the content type is specified by **target-type** attribute.
- sound-recorder: MMS Client invokes sound recorder application. The application is expected to return Speech Audio as defined in [MMSCONF] and the content type is specified by **target-type** attribute.
- address-book: MMS Client invokes address book. The application is expected to return vCard[OMA-vObjectOMAProfile-V1.0] that contains N, Version, and ADDR as defined in [MMSCONF] if there is no parameter. The content type returned by the application is specified by **target-type** attribute as “text/x-vcard”.

- file-manager: MMS Client invokes file manager. The application is expected to return a file with content type as specified by **target-type** attribute.

The **step** element MAY have a **param** attribute value to pass arguments to the called application. The **param** attribute SHALL contain **xs:string** in the manner of query-string description in [RFC2396]. The device application SHOULD support such parameters, but the standardization of them is out of scope of this specification.

*Note1: If the above applications return an unexpected result then the client should handle it appropriately.*

*Note2: MMS Template author should be wary of limited MMS Client resources to avoid an excessive call of device applications.*

*Note3: Some common device parameters should be standardized to improve IOP.*

The **target-name** attribute identifies the name of media object that will be replaced by this step. The **target-name** attribute SHALL contain an **xs:string** value. The media object returned by invoked application will replace the target media object. If **target-name** attribute specifies a nonexistent media object, the returned media object from invoked application will be just added to the list of objects being collected with no actual replacement step occurring.When creating Resulting MM, the references of new media objects in the presentation language SHALL be updated correctly.

MMS Postcard Conformance: If **target-name** attribute value is specified as “X-MMS-GREETINGTEXT”, it means that the text specified by the **target-name** attribute is the greeting text for Postcard service that is defined in section 17 of MMS Postcard Conformance of [MMSCONF]. The media object SHALL be encoded appropriately, as described in section 17.2 of [MMSCONF].

The **target-type** attribute identifies the content type of the returned media object. The invoked application SHALL return a media object encoded in specified content type. The **target-type** attribute SHALL contain an **xs:string** value. The **target-type** attribute is specified by a comma-separated list of media types, as per [RFC2045]. The specified media types SHALL belong to same media object group that can be referenced by same SMIL media object element.

*Note: When MMS Client can not satisfy media object request, it MAY skip the input.*

When the handset does not support the specified application, MMS Client MAY invoke file manager to allow user to select a file.

The **step** element MAY have a **required** attribute. When present, the **required** attribute SHALL contain a **xs:boolean** value. A "true" value indicates that the input of media object specified by this **step** element is mandatory. A "false" value indicates that it is optional. If the attribute is not specified, the effect is as if a value of "false" were specified.

## 5.2.2 MMS Client

If the MMS Client implements MMS Templates then the following requirements SHALL be followed.

For the sake of clarity, the requirements are partitioned according to the reference-processing model. It is not required that an implementation execute each phase atomically.

### 5.2.2.1 Phase 1: Identify Needed Resources

MMS Client SHOULD validate that the MTD is valid. If not, MMS Client SHALL ignore it.

If the version of MTD is not supported by MMS Client, MMS Client SHALL NOT go to the next phase.

MMS Client SHALL be able to parse all MTD elements and attributes. If unknown elements or attributes are encountered they SHOULD be ignored.

MMS Client SHALL use the MTD to construct and execute an MMS Message Template.

MMS Client SHALL process MTD elements and attributes which are specified as "M" in table 1 with the exceptions mentioned in parenthesis.

MMS Client SHOULD process MTD elements and attributes which are specified as "O" in table 1.

*Note: The process of meta information module elements is to provide a method to allow user to access the information.*

| module | elements | attributes | support |
|---|---|---|---|
| Architecture | mmstemplate | | M |
| | head | | M |
| | | xml:lang | O |
| | body | | M |
| | | xml:lang | O |
| Meta Information | title | | M |
| | | xml:lang | O |
| | description | | O |
| | | xml:lang | O |
| | author | | O |
| | | xml:lang | O |
| | date | | O |
| | version | | O |
| | meta | | O |
| | | content | O |
| | | name | O |
| Message | message | | M |
| | | xml:lang | O |
| | to-header | | M |
| | | editable | M |
| | cc-header | | M |
| | | editable | M |
| | subject-header | | M |
| | | xml:lang | O |
| | | editable | M |
| Wizard | wizard | | M (If Wizard part is supported) |
| | | xml:lang | O |
| | step | | M (If Wizard part is supported) |
| | | xml:lang | O |
| | | guide | O |
| | | app | O |

| | | | |
|---|---|---|---|
| | | **param** | O |
| | | **target-name** | M (If Wizard part is supported) |
| | | **target-type** | M (If Wizard part is supported) |
| | | **required** | M (If Wizard part is supported) |

**Table 1: MMS Message Template language client conformance**

#### 5.2.2.2 Phase 2: Acquire Replaceable Media Objects

If the MMS Client does not support the Wizard function, then it is expected that there is functionality implemented to appropriately handle replaceable media objects. This functionality is part of Phase 2 since replacement functionality SHALL occur by Wizard or some other mechanism.

The requirements described below are only necessary if the MMS Client supports the Wizard function.

MMS Client SHALL provide a method to input appropriate media objects that are specified in **step** elements.

MMS Client SHALL prohibit to change fixed media objects that are not specified in any **step** elements.

MMS Client SHALL support the following applications to input media objects.

- ✓ Plain text editor
- ✓ File manger
- ✓ Address book

MMS Client SHOULD support the following applications to input media objects.

- ✓ Still-camera application
- ✓ Video-camera application
- ✓ Sound recorder application
- ✓ Rich text editor

MMS Client SHOULD invoke specified application to input media object as specified in **step** elements.

MMS Client SHOULD pass the parameters specified in the **param** attribute to the application.

MMS Client SHOULD display guide message as specified in **step** elements.

MMS Client SHOULD support backward/forward navigation between several steps while maintaining user input data.

MMS Client SHALL NOT be allowed to modify/delete fixed media objects that are not referenced by any steps in wizard.

When MMS Client completes processing all steps, MMS Client SHALL move to next phase.

#### 5.2.2.3 Phase 3: Create Resulting MM

MMS Client SHOULD support backward navigation to phase 2.

MMS Client SHALL be able to generate a Resulting MM from the MMS presentation part, fixed media objects and user-specified media objects.

MMS Client SHALL set MMS header values of the Resulting MM as specified in message part.

MMS Client SHALL make pre-filled MMS header values of the resulting MM available to the user.

If MMS Client supports Wizard function, it SHALL check required media objects are appropriately inputted with specified media type in **target-type** attributes. If not, MMS Client SHALL NOT send the Resulting MM.

The Resulting MM SHALL be compliant to the Core MM Content Domain as defined in [MMSCONF].

## 5.3 MMS Message Template Distribution (Informative)

### 5.3.1 MTD

MMS Client can obtain new MMS Message Templates via MMS or OMA download. In the distribution of templates, MMS Message Template (all needed resources containing MTD, MMS presentation part, and media objects) is delivered to the handset.

The handset stores several MMS Message Templates in its internal storage. Meta information described in the MTD such as title or description helps to manage these templates.

When MMS Client receives an unsupported version of MTD, it SHALL ignore it.

### 5.3.2 Resulting MM

Resulting MM is compliant to the Core MM Content Domain defined in [MMSCONF].

Media objects derived from service provider may be protected by separate delivery method DRM as defined in [OMADRM].

# 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 1.3 History

| Document Identifier | Date | Sections | Description |
|---|---|---|---|
| Draft Versions<br>OMA-MMS-Messae-Template-Specification-V1_3 | 14 Dec. 2004 | | 1st draft |
| | 03 Mar 2005 | | Inclusion of CRs and editorial changes<br>The contents of these documents were included:<br>OMA-MMSG-2005-0030<br>OMA-MMSG-2005-0054R04<br>OMA-MMSG-2005-0055R03<br>OMA-MMSG-2005-0056R03<br>OMA-MMSG-2005-0057R03<br>OMA-MMSG-2005-0059R02<br>OMA-MMSG-2005-0065R01<br>OMA-MMSG-2005-0066 |
| | 22 Mar 2005 | | Inclusion of CRs and editorial changes<br>The contents of these documents were included:<br>OMA-MMSG-2005-0067R02<br>OMA-MMSG-2005-0101R01<br>OMA-MMSG-2005-0065R02 |
| | 06 May 2005 | | Changes made per OMA-MMSG-2005-0130R02-CONRR-MMS-V1_3-20050422-D<br>Review Report |
| Draft versions<br>OMA-TS-MMS-TEMP-V1_3 | 23 May 2005 | | Changes made per CR: OMA-MMSG-2005-0144-Security-changes-to-meta-element |
| Candidate versions<br>OMA-TS-MMS-TEMP-V1_3 | 17 Jun 2005 | n/a<br><br><br>All | Status changed to Candidate by TP<br>TP ref # OMA-TP-2005-0190R03-MMS-V1_3-for-Candidate-approval.<br>General editorial clean-up of styles for publication. |
| | 27 Oct 2005 | 5.2.1.5<br>2.1 | Inclusion of CR: OMA-MMSG-2005-0222R01.<br>Editorial change to fix broken link. |
| Draft Versions<br>OMA-TS-MMS-TEMP-V1_3 | 31 Aug 2006 | 2.1, 3.3, 5.1, 5.2.1, App. B (removed), B.1 (formerly C.1). | Class 2/3 CRs incorporated:<br>OMA-MMSG-2006-0015<br>OMA-MMSG-2006-0065R01<br>OMA-MMSG-2006-0072R01 |
| Candidate versions<br>OMA-TS-MMS-TEMP-V1_3 | 28 Jan 2008 | All | Status changed to Candidate by TP<br>TP ref # OMA-TP-2007-0494-INP_MMS_V1_3_for_Candidate_ReApproval. |

# Appendix B. Static Conformance Requirements (Normative)

## B.1 MTD Document Conformance

| Item | Function | Reference | Status | Requirement |
|---|---|---|---|---|
| MMSTEMP-MTD-C-001 | Conform to the constraints expressed in the XML Schema found in [MMSMTDSCHEMA]. | 5.2.1 | M | MMSTEMP-MTD-C-002AND MMSTEMP-MTD-C-003AND MMSTEMP-MTD-C-004 |
| MMSTEMP-MTD-C-002 | The MTD namespace name | 5.2.1 | M | |
| MMSTEMP-MTD-C-003 | Exclusively for the use of DTD; the DOCTYPE declaration in the document prior to the root element, with the public identifier | 5.2.1 | M | |
| MMSTEMP-MTD-C-004 | Exclusively for the use of DTD; the public identifier referenced in the DTD found in [MMSMTDDTD] | 5.2.1 | M | |

**Table 2: MTD requirements**

## B.2 MMS Template Client conformance

| Item | Function | Reference | Status | Requirement |
|---|---|---|---|---|
| MMSTEMP-MMSTC-C-001 | Validate MTD | 5.2.2.1 | O | |
| MMSTEMP-MMSTC-C-002 | Ignore MTD if it is not valid. | 5.2.2.1 | M | |
| MMSTEMP-MMSTC-C-003 | Identify MTD Version | 5.2.2.1 | M | |
| MMSTEMP-MMSTC-C-004 | Terminate if the version of MTD is not supported | 5.2.2.1 | M | |
| MMSTEMP-MMSTC-C-005 | Be able to parse all MTD elements and attributes | 5.2.2.1 | M | |
| MMSTEMP-MMSTC-C-006 | Ignore unknown elements and attributes | 5.2.2.1 | O | |
| MMSTEMP-MMSTC-C-007 | Use MTD to construct and execute MMS Template | 5.2.2.1 | M | |
| MMSTEMP-MMSTC-C-008 | Process MTD elements and attributes which are specified as "M" in Table 1 with the exceptions mentioned in parenthesis | 5.2.2.1 | M | |
| MMSTEMP-MMSTC-C-009 | Process MTD elements and attributes which are specified as "O" in Table 1 | 5.2.2.1 | O | |
| MMSTEMP-MMSTC-C-010 | Media Object Replacement functionality | 5.2.2.2 | O | |
| MMSTEMP-MMSTC-C-011 | Support Wizard Part | 5.2.2.2 | O | C.3 Wizard Conformance (MMSTEMP-WIZC-C-001 AND MMSTEMP-WIZC-C-002 AND MMSTEMP-WIZC-C-003 AND MMSTEMP-WIZC-C-004 AND MMSTEMP-WIZC-C-005 AND MMSTEMP-WIZC-C-006 AND MMSTEMP-WIZC-C-007 AND MMSTEMP-WIZC-C-008 |
| MMSTEMP-MMSTC-C-012 | Support backward navigation to phase 2 | 5.2.2.3 | O | |
| MMSTEMP-MMSTC-C-013 | Generate a Resulting MM | 5.2.2.3 | M | |
| MMSTEMP-MMSTC-C-014 | Set MMS header values of Resulting MM | 5.2.2.3 | M | |
| MMSTEMP-MMSTC-C-015 | Make pre-filled MMS header values of the Resulting MM available to the user | 5.2.2.3 | M | |
| MMSTEMP-MMSTC-C-016 | Resulting MM be compliant to the Core MM Content Domain | 5.2.2.3 | M | |

**Table 3: MMS Client requirements**

## B.3 Wizard conformance

| Item | Function | Reference | Status | Requirement |
|---|---|---|---|---|
| MMSTEMP-WIZC-C-001 | Replace media objects specified by target name | 5.2.2.2 | M | |
| MMSTEMP-WIZC-C-002 | Prohibit replacement with media object violating target type | 5.2.2.2 | M | |
| MMSTEMP-WIZC-C-003 | Prohibit changing fixed media objects not specified in any step elements | 5.2.2.2 | M | |
| MMSTEMP-WIZC-C-004 | Show Guide message | 5.2.2.2 | O | |
| MMSTEMP-WIZC-C-005 | Invoke specified applications to input media objects specified by step elements | 5.2.2.2 | O | |
| MMSTEMP-WIZC-C-006 | Pass the parameters to the application | 5.2.2.2 | O | |
| MMSTEMP-WIZC-C-007 | Support backward/forward navigation between several steps | 5.2.2.2 | O | |
| MMSTEMP-WIZC-C-008 | Check media objects indicated by required attribute are appropriately inputted with specified media type in step elements. | 5.2.2.3 | M | |

**Table 4: Wizard Part requirements**