

[MS-OXOSMMS]: Short Message Service (SMS) and Multimedia Messaging Service (MMS) Object Protocol Specification

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft's Open Specification Promise (available here: <http://www.microsoft.com/interop/osp>) or the Community Promise (available here: <http://www.microsoft.com/interop/cp/default.mspx>). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.
- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

Revision Summary

Date	Revision History	Revision Class	Comments
04/04/2008	0.1		Initial Availability.
04/25/2008	0.2		Revised and updated property names and other technical content.
06/27/2008	1.0		Initial Release.
08/06/2008	1.01		Revised and edited technical content.
09/03/2008	1.02		Updated references.
12/03/2008	1.03		Updated IP notice.
04/10/2009	2.0		Updated applicable product releases.
07/15/2009	3.0	Major	Revised and edited for technical content.
11/04/2009	4.0.0	Major	Updated and revised the technical content.
02/10/2010	4.0.0	None	Version 4.0.0 release
05/05/2010	5.0.0	Major	Updated and revised the technical content.
08/04/2010	5.1	Minor	Clarified the meaning of the technical content.
11/03/2010	5.2	Minor	Clarified the meaning of the technical content.
03/18/2011	5.2	No change	No changes to the meaning, language, or formatting of the technical content.
08/05/2011	5.3	Minor	Clarified the meaning of the technical content.
10/07/2011	6.0	Major	Significantly changed the technical content.

Table of Contents

1 Introduction	5
1.1 Glossary	5
1.2 References.....	5
1.2.1 Normative References.....	5
1.2.2 Informative References	6
1.3 Overview	6
1.4 Relationship to Other Protocols.....	6
1.5 Prerequisites/Preconditions	6
1.6 Applicability Statement.....	7
1.7 Versioning and Capability Negotiation.....	7
1.8 Vendor-Extensible Fields.....	7
1.9 Standards Assignments	7
2 Messages	8
2.1 Transport.....	8
2.2 Message Syntax	8
2.2.1 Common SMS and MMS Object Properties	8
2.2.1.1 PidNameOMSAccountGuid Property	8
2.2.1.2 PidNameOMSScheduleTime Property	8
2.2.1.3 PidNameOMSServiceType Property	8
2.2.1.4 PidNameOMSSourceType Property	8
2.2.1.5 PidNameContentClass Property.....	9
2.2.1.6 PidNameOMSMobileModel Property	9
2.2.2 Additional Property Constraints	9
2.2.2.1 PidTagIconIndex Property	9
2.2.2.2 PidTagMessageClass Property.....	9
2.2.2.3 Body Properties	10
2.2.2.4 PidTagNormalizedSubject Property.....	10
3 Protocol Details	11
3.1 Common Details	11
3.1.1 Abstract Data Model	11
3.1.2 Timers	11
3.1.3 Initialization	11
3.1.4 Higher-Layer Triggered Events.....	11
3.1.4.1 Creation of an SMS or MMS Object.....	11
3.1.4.2 Modification of an SMS or MMS Object.....	11
3.1.4.3 Deletion of an SMS or MMS Object.....	11
3.1.5 Message Processing Events and Sequencing Rules.....	11
3.1.6 Timer Events	11
3.1.7 Other Local Events	11
4 Protocol Examples	12
4.1 Sample SMS Object	12
4.2 Sample MMS Object	13
5 Security	17
5.1 Security Considerations for Implementers.....	17
5.2 Index of Security Parameters	17
6 Appendix A: Product Behavior	18

7 Change Tracking.....	19
8 Index	21

1 Introduction

The SMS and MMS Object protocol defines properties of objects that model **Short Message Service (SMS)** and **Multimedia Messaging Service (MMS)** messages.

Sections 1.8, 2, and 3 of this specification are normative and contain RFC 2119 language. Sections 1.5 and 1.9 are also normative but cannot contain RFC 2119 language. All other sections and examples in this specification are informative.

1.1 Glossary

The following terms are defined in [\[MS-GLOS\]](#):

Coordinated Universal Time (UTC)
flags
GUID
handle
Unicode

The following terms are defined in [\[MS-OXGLOS\]](#):

Attachment object
calendar
Folder object
Hypertext Markup Language (HTML)
Message object
Multimedia Messaging Service (MMS)
named property
plain text
property ID
reminder
remote operation (ROP)
rule
Short Message Service (SMS)
store
stream

The following terms are specific to this document:

Multimedia Messaging Service (MMS) object: A Message object that represents a Multimedia Messaging Service (MMS) message in a messaging store.

SMS object: A Message object that represents an Short Message Service (SMS) message in a messaging store.

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as described in [\[RFC2119\]](#). All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We

will assist you in finding the relevant information. Please check the archive site, <http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624>, as an additional source.

[MS-OXCDATA] Microsoft Corporation, "[Data Structures](#)".

[MS-OXCMAIL] Microsoft Corporation, "[RFC2822 and MIME to E-Mail Object Conversion Algorithm](#)".

[MS-OXCMSG] Microsoft Corporation, "[Message and Attachment Object Protocol Specification](#)".

[MS-OXCPRPT] Microsoft Corporation, "[Property and Stream Object Protocol Specification](#)".

[MS-OXOMSG] Microsoft Corporation, "[E-Mail Object Protocol Specification](#)".

[MS-OXPROPS] Microsoft Corporation, "[Exchange Server Protocols Master Property List](#)".

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.rfc-editor.org/rfc/rfc2119.txt>

1.2.2 Informative References

[MS-GLOS] Microsoft Corporation, "[Windows Protocols Master Glossary](#)".

[MS-OXCROPS] Microsoft Corporation, "[Remote Operations \(ROP\) List and Encoding Protocol Specification](#)".

[MS-OXGLOS] Microsoft Corporation, "[Exchange Server Protocols Master Glossary](#)".

[SMIL2.1] Bulterman, D., Grassel, G., Jansen, J., Eds., et al., "Synchronized Multimedia Integration Language (SMIL 2.1)", W3C Recommendation, December 2005, <http://www.w3.org/TR/2005/REC-SMIL2-20051213/>

1.3 Overview

The SMS and MMS Object protocol specifies the representation of SMS text messages and MMS multimedia messages in a messaging **store**. This protocol extends the Message and Attachment Object protocol in that it defines new properties and adds restrictions to the properties that are specified in [\[MS-OXCMSG\]](#).

This document specifies the properties that are unique to **SMS objects** and **MMS objects**. An SMS object is characterized by a short unformatted text body. An MMS object is characterized by text and multimedia components. SMS and MMS objects are stored in **Folder objects**. The SMS and MMS Object protocol also specifies how an SMS or MMS object is created and manipulated.

1.4 Relationship to Other Protocols

This protocol has the same dependencies as the Message and Attachment Object protocol, which it extends. For more details about the Message and Attachment Object protocol, see [\[MS-OXCMSG\]](#).

1.5 Prerequisites/Preconditions

This protocol has the same prerequisites and preconditions as the Message and Attachment Object protocol, as specified in [\[MS-OXCMSG\]](#).

1.6 Applicability Statement

None.

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

This protocol provides no vendor-extensibility beyond what is already specified in [\[MS-OXCMMSG\]](#).

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

The SMS and MMS Object protocol uses the protocols defined in [\[MS-OXCPRPT\]](#) and [\[MS-OXCMSG\]](#) as its primary transport mechanism.

2.2 Message Syntax

SMS and MMS objects can be created and modified by clients and servers. Except where noted below, this section defines constraints under which both clients and servers operate.

Clients operate on SMS and MMS objects using the Message and Attachment Object protocol, as specified in [\[MS-OXCMSG\]](#). How a server operates on SMS and MMS objects is implementation-dependent. The results of any such operations are exposed to clients in a manner that is consistent with this protocol.

Unless otherwise specified, SMS and MMS objects adhere to all property constraints specified in [\[MS-OXPROPS\]](#) and [\[MS-OXCMSG\]](#). SMS and MMS objects can also contain other properties, which are specified in [\[MS-OXPROPS\]](#), but these properties have no impact on this protocol.

2.2.1 Common SMS and MMS Object Properties

2.2.1.1 PidNameOMSAccountGuid Property

Type: **PtypString** ([\[MS-OXCADATA\]](#) section 2.11.1)

The **PidNameOMSAccountGuid** property ([\[MS-OXPROPS\]](#) section 2.503) encodes the **GUID** of the SMS account used to deliver the message in the following format (including the braces): {DWORD-WORD-WORD-WORD.DWORD}; for example, "{c200e360-38c5-11ce-ae62-08002b2b79ef}".

2.2.1.2 PidNameOMSScheduleTime Property

Type: **PtypTime**, in **UTC** ([\[MS-OXCADATA\]](#) section 2.11.1)

The **PidNameOMSScheduleTime** property ([\[MS-OXPROPS\]](#) section 2.505) specifies the time at which the client requests that the service provider sends the SMS or MMS message.

2.2.1.3 PidNameOMSServiceType Property

Type: **PtypInteger32** ([\[MS-OXCADATA\]](#) section 2.11.1)

The **PidNameOMSServiceType** property ([\[MS-OXPROPS\]](#) section 2.506) indicates the type of service used to send the SMS or MMS message; MUST be one of the values in the following table.

Value	Meaning
0x00000001	SMS
0x00000004	MMS

2.2.1.4 PidNameOMSSourceType Property

Type: **PtypInteger32** ([\[MS-OXCADATA\]](#) section 2.11.1)

The **PidNameOMSSourceType** property ([\[MS-OXPROPS\]](#) section 2.507) indicates the source of the SMS or MMS message; MUST be one of the values in the following table.

Value	Source type
0x00000000	XMS Inspector
0x00000001	Reminder
0x00000002	Calendar Summary
0x00000003	Rule (1)
0x00000004	Unknown

2.2.1.5 PidNameContentClass Property

Type: **PtypString** ([\[MS-OXCDATA\]](#) section 2.11.1)

The **PidNameContentClass** property ([\[MS-OXPROPS\]](#) section 2.428) MUST be set to one of the values in the following table.

Value	Meaning
MS-OMS-SMS	SMS
MS-OMS-MMS	MMS

2.2.1.6 PidNameOMSMobileModel Property

Type: **PtypString** ([\[MS-OXCDATA\]](#) section 2.11.1)

The **PidNameOMSMobileModel** property ([\[MS-OXPROPS\]](#) section 2.504) indicates the model of the mobile device used to send the SMS or MMS message.

2.2.2 Additional Property Constraints

This protocol specifies additional constraints on the following properties beyond what is specified in [\[MS-OXCMSG\]](#) and [\[MS-OXOMSG\]](#).

2.2.2.1 PidTagIconIndex Property

Type: **PtypInteger32** ([\[MS-OXCDATA\]](#) section 2.11.1)

The **PidTagIconIndex** property ([\[MS-OXPROPS\]](#) section 2.793) specifies which icon is to be used by a user interface when displaying a group of SMS and/or MMS objects; SHOULD be set; [<1>](#) if set, MUST be "0xFFFFFFFF".

2.2.2.2 PidTagMessageClass Property

Type: **PtypString**, case-insensitive ([\[MS-OXCDATA\]](#) section 2.11.1)

The **PidTagMessageClass** property ([\[MS-OXPROPS\]](#) section 2.843) specifies the type of the **Message object**. In addition to meeting the criteria specified in [\[MS-OXCMSG\]](#) section 2.2.1.3, this value MUST be set to "IPM.Note.Mobile.SMS" or begin with "IPM.Note.Mobile.SMS" for SMS objects, or MUST be set to "IPM.Note.Mobile.MMS" or begin with "IPM.Note.Mobile.MMS." for MMS objects.

2.2.2.3 Body Properties

The contents of SMS objects are stored and retrieved following the details specified for **plain text** messages in [\[MS-OXCMAIL\]](#) section 2.1.3.3.1.<2>

The contents of MMS objects are stored and retrieved following the details specified for **HTML** messages, as specified in [\[MS-OXCMAIL\]](#) section 2.1.3.3.1.<3>

2.2.2.4 PidTagNormalizedSubject Property

Type: **PtypString** ([\[MS-OXCDATA\]](#) section 2.11.1)

The **PidTagNormalizedSubject** property ([\[MS-OXPROPS\]](#) section 2.864) contains an abbreviated version of the contents of the message suitable for displaying groups of SMS objects to a user. For MMS objects, only the constraints specified in [\[MS-OXCMSG\]](#) section 2.2.1.10 apply.

3 Protocol Details

3.1 Common Details

The client and server roles create and operate on SMS and MMS objects; otherwise, the roles operate as specified in [\[MS-OXCMSG\]](#) section 3.1 and [\[MS-OXCMSG\]](#) section 3.2.

3.1.1 Abstract Data Model

None.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Higher-Layer Triggered Events

3.1.4.1 Creation of an SMS or MMS Object

To create an SMS or MMS object, the server or client sets properties in accordance with the requirements in section 2 and [\[MS-OXCPRPT\]](#), and saves the resulting Message object as specified in [\[MS-OXCMSG\]](#).

3.1.4.2 Modification of an SMS or MMS Object

When modifying an SMS or MMS object, the client or server modifies any of the properties in accordance with the requirements in section 2 and [\[MS-OXCPRPT\]](#), and saves the Message object as specified in [\[MS-OXCMSG\]](#).

3.1.4.3 Deletion of an SMS or MMS Object

An SMS or MMS object has no special deletion semantics beyond what is specified in [\[MS-OXCMSG\]](#).

3.1.5 Message Processing Events and Sequencing Rules

None.

3.1.6 Timer Events

None.

3.1.7 Other Local Events

None.

4 Protocol Examples

4.1 Sample SMS Object

A user creates an SMS object, types in some text, and sends it. The following is a description of what a client might do to accomplish the user's intentions and the responses a server might return. For more details about **remote operations (ROPs)**, see [\[MS-OXCPRPT\]](#) and [\[MS-OXCMSG\]](#).

Before manipulating SMS objects, the client needs to ask the server to perform a mapping from **named properties** to **property IDs**, using the **RopGetPropertyIDsFromNames** ROP ([\[MS-OXCROPS\]](#) section 2.2.8.1).

Property	Property set GUID	NameID
PidNameOMSMobileModel ([MS-OXPROPS] section 2.504)	{00020329-0000-0000-C00000000046}	OMSMobileModel
PidNameOMSAccountGuid ([MS-OXPROPS] section 2.503)	{00020329-0000-0000-C00000000046}	OMSAccountGuid
PidNameOMSServiceType ([MS-OXPROPS] section 2.506)	{00020329-0000-0000-C00000000046}	OMSServiceType
PidNameOMSSourceType ([MS-OXPROPS] section 2.507)	{00020329-0000-0000-C00000000046}	OMSSourceType

The server might respond with the following identifiers, which are used in the example that follows. (The actual identifiers are at the discretion of the server.)

Property	Property ID
PidNameOMSMobileModel	0x84c3
PidNameOMSAccountGuid	0x84c4
PidNameOMSServiceType	0x84c5
PidNameOMSSourceType	0x84c6

To create an SMS object, the client uses the **RopCreateMessage** ROP ([\[MS-OXCROPS\]](#) section 2.2.6.2). In response, the server returns a success code and a **handle** to a Message object.

After the user input his content for the SMS object, the client uses the **RopSetProperties** ROP ([\[MS-OXCROPS\]](#) section 2.2.8.6) to transmit his data to the server.

Property	Property ID	Data type	Value
PidNameOMSAccountGuid	0x84c4	0x001f (PtypString ([MS-OXCADATA] section 2.11.1))	{01234567-0123-0123-0123-0123456789ab}
PidNameOMSMobileModel	0x84c3	0x001f (PtypString)	(null)
PidNameOMSServiceType	0x84c5	0x0003 (PtypInteger32 ([MS-OXCADATA] section 2.11.1))	0x00000001

Property	Property ID	Data type	Value
		2.11.1))	
PidNameOMSSourceType	0x84c6	0x0003 (PtypInteger32)	0x00000000
PidTagBody ([MS-OXPROPS] section 2.676)	0x1000	0x001f (PtypString)	What time is the meeting?
PidTagInternetCodepage ([MS-OXPROPS] section 2.803)	0x3fde	0x0003 (PtypInteger32)	0x0000FDE9
PidTagMessageClass ([MS-OXPROPS] section 2.843)	0x001a	0x001a 0x001f (PtypString)	IPM.Note.Mobile.SMS
PidTagNormalizedSubject ([MS-OXPROPS] section 2.864)	0x0e1d	0x001f (PtypString)	What time is the meeting?
PidTagSubjectPrefix ([MS-OXPROPS] section 2.1087)	0x003d	0x001f (PtypString)	(null)

When the user is ready to send their message message, the client uses the **RopSaveChangesMessage** ROP ([\[MS-OXCROPS\]](#) section 2.2.6.3) to commit the properties on the server, and then the **RopRelease** ROP ([\[MS-OXCROPS\]](#) section 2.2.15.3) to release the SMS object. The client then submits the message to an SMS provider using an appropriate messaging protocol.

The values of some properties change during the execution of the **RopSaveChangesMessage** ROP, but the properties defined in this specification do not change.

4.2 Sample MMS Object

A user creates an MMS object, gives it a subject, types in some text, attaches a picture, and sends it. The following is a description of what a client might do to accomplish the users intentions and the responses a server might return. For more details about ROPs, see [\[MS-OXCPRPT\]](#) and [\[MS-OXCMSG\]](#).

Before manipulating an MMS object, the client needs to ask the server to perform a mapping from named properties to property IDs, using the **RopGetPropertyIDsFromNames** ROP ([\[MS-OXCROPS\]](#) section 2.2.8.1).

Property	Property set GUID	NameID
PidNameOMSMobileModel ([MS-OXPROPS] section 2.504)	{00020329-0000-0000-C00000000046}	OMSMobileModel
PidNameOMSAccountGuid ([MS-OXPROPS] section 2.503)	{00020329-0000-0000-C00000000046}	OMSAccountGuid
PidNameOMSServiceType ([MS-OXPROPS] section 2.506)	{00020329-0000-0000-C00000000046}	OMSServiceType
PidNameOMSSourceType ([MS-OXPROPS] section 2.507)	{00020329-0000-0000-C00000000046}	OMSSourceType

The server might respond with the following identifiers, which are used in the example that follows. (The actual identifiers are at the discretion of the server.)

Property	Property ID
PidNameOMSMobileModel	0x84ce
PidNameOMSAccountGuid	0x84cf
PidNameOMSServiceType	0x84d0
PidNameOMSSourceType	0x84d1

To create an MMS object, the client uses the **RopCreateMessage** ROP ([\[MS-OXCROPS\]](#) section 2.2.6.2). The server returns a success code and a handle to an object.

After the user input their content for the MMS object, the client uses the **RopSetProperties** ROP ([\[MS-OXCROPS\]](#) section 2.2.8.6) to transmit the data to the server.

Property	Property ID	Data type	Value
PidNameOMSAccountGuid	0x84cf	0x001f (PtypString ([MS-OXCADATA] section 2.11.1))	{01234567-0123-0123-0123456789abc}
PidNameOMSMobileModel	0x84ce	0x001f (PtypString)	(empty)
PidNameOMSServiceType	0x84d0	0x0003 (PtypInteger32 ([MS-OXCADATA] section 2.11.1))	0x00000004
PidNameOMSSourceType	0x84d1	0x0003 (PtypInteger32)	0x00000000
PidTagInternetCodepage ([MS-OXPROPS] section 2.803)	0x3fde	0x0003 (PtypInteger32)	0x0000FDE9
PidTagHtml ([MS-OXPROPS] section 2.789)	0x1013	0x0102 (PtypBinary ([MS-OXCADATA] section 2.11.1))	See below
PidTagIconIndex ([MS-OXPROPS] section 2.793)	0x1080	0x0003 (PtypInteger32)	0xFFFFFFFF
PidTagMessageClass ([MS-OXPROPS] section 2.843)	0x001a	0x001a 0x001f (PtypString)	IPM.Note.Mobile.MMS
PidTagMessageFlags ([MS-OXPROPS] section 2.847)	0x0e07	0x0003 (PtypInteger32)	Flags: 0x00000018 MSGFLAG_UNSENT MSGFLAG_HASATTACH
PidTagNormalizedSubject ([MS-OXPROPS] section 2.864)	0x0e1d	0x001f (PtypString)	Here's the photo.
PidTagSubjectPrefix ([MS-OXPROPS] section 2.1087)	0x003d	0x001f (PtypString)	(empty)

The **PidTagHtml** property is a binary property containing the following text.

```
<HTML>
```

```

<BODY>
<IMG SRC="cid:Att1.jpg@AB1B43B2B0594564.B94EF7ABB12B49BA" border="0">
<BR>
This is the photo you asked for.
<BR>
<A HREF="cid:Att0.txt@AB1B43B2B0594564.B94EF7ABB12B49BA"></A>
</BODY>
</HTML>

```

The client uses the **RopCreateAttachment** ROP ([\[MS-OXCROPS\]](#) section 2.2.6.13) to allocate space for a data file in the message. In response, the server returns a success code and a handle to an **Attachment object**. The client then uses this handle with the **RopSetProperties** ROP ([\[MS-OXCROPS\]](#) section 2.2.8.6) to transmit data about the attachment to the server.

Property	Property ID	Data type	Value
PidTagAttachmentHidden ([MS-OXPROPS] section 2.657)	0x7ffe	0x000b (PtypBoolean ([MS-OXCDATA] section 2.11.1))	0x01
PidTagAttachMethod ([MS-OXPROPS] section 2.659)	0x3705	0x0003 (PtypInteger32)	0x00000001 (ATTACH_BY_VALUE)
PidTagAttachContentId ([MS-OXPROPS] section 2.645)	0x3712	0x001f (PtypString)	MMS.smil@AB1B43B2B0594564.B94EF7ABB12B49BA
PidTagAttachMimeType ([MS-OXPROPS] section 2.660)	0x370e	0x001f (PtypString)	application/smil
PidTagAttachLongFilename ([MS-OXPROPS] section 2.653)	0x3707	0x001f (PtypString)	MMS.smil

The client sets the contents of the attachment by using the attachment handle with the **RopOpenStream** ROP ([\[MS-OXCROPS\]](#) section 2.2.9.1), passing in the **PidTagAttachDataBinary** property ([\[MS-OXPROPS\]](#) section 2.647) as the property to open. With the handle returned from the **RopOpenStream** ROP, the client calls the **RopWriteStream** ROP ([\[MS-OXCROPS\]](#) section 2.2.9.3), writing out the contents of the Synchronized Multimedia Integration Language (SMIL) file, the format of which is described in [\[SMIL2.1\]](#), describing the layout of the MMS message. The client follows this with the **RopRelease** ROP ([\[MS-OXCROPS\]](#) section 2.2.15.3) on the **stream (2)** handle, then the **RopSaveChangesAttachment** ROP ([\[MS-OXCROPS\]](#) section 2.2.6.15) to commit the changes, and the **RopRelease** ROP to release the handle to the attachment.

The client repeats the process from the **RopCreateAttachment** ROP ([\[MS-OXCROPS\]](#) section 2.2.6.13) for the **RopRelease** ROP with the attachment handle twice more, once for a plain-text version of the body, and once for the image. The attachment containing the body uses the following properties and values with the **RopSetProperties** ROP.

Property	Property ID	Data type	Value
PidTagAttachmentHidden	0x7ffe	0x000b (PtypBoolean)	0x01
PidTagAttachMethod	0x3705	0x0003 (PtypInteger32)	0x00000001 (ATTACH_BY_VALUE)
PidTagAttachContentId	0x3712	0x001f (PtypString)	Att0.txt@AB1B43B2B0594564.B94EF7ABB12B49BA
PidTagAttachMimeTag	0x370e	0x001f (PtypString)	text/plain
PidTagAttachLongFilename	0x3707	0x001f (PtypString)	1.txt

The **RopOpenStream** ROP for the plain-text body is also on the **PidTagAttachDataBinary** property, but the contents written are **Unicode** text. The last attachment the client creates contains the image, and the **RopSetProperties** ROP sends the following data.

Property	Property ID	Data type	Value
PidTagAttachmentHidden	0x7ffe	0x000b (PtypBoolean)	0x01
PidTagAttachMethod	0x3705	0x0003 (PtypInteger32)	0x00000001 (ATTACH_BY_VALUE)
PidTagAttachContentId	0x3712	0x001f (PtypString)	Att1.jpg@AB1B43B2B0594564.B94EF7ABB12B49BA
PidTagAttachMimeTag	0x370e	0x001f (PtypString)	image/jpeg
PidTagAttachLongFilename	0x3707	0x001f (PtypString)	photo.jpg

The contents of the **PidTagAttachDataBinary** property on the image attachment are the binary contents of the image file.

When the user is ready to send their message, the client uses the **RopSaveChangesMessage** ROP ([\[MS-OXCROPS\]](#) section 2.2.6.3) to commit the properties on the server, and then the **RopRelease** ROP to release the MMS object. The client then submits the message to an MMS provider using an appropriate messaging protocol.

The values of some properties change during the execution of the **RopSaveChangesMessage** ROP, but the properties specified in this protocol do not change.

5 Security

5.1 Security Considerations for Implementers

There are no special security considerations specific to the SMS and MMS Object protocol. General security considerations pertaining to the underlying transport apply, as specified in [\[MS-OXCMSG\]](#) and [\[MS-OXCPRPT\]](#).

5.2 Index of Security Parameters

None.

6 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Microsoft® Exchange Server 2003
- Microsoft® Exchange Server 2007
- Microsoft® Exchange Server 2010
- Microsoft® Office Outlook® 2003
- Microsoft® Office Outlook® 2007
- Microsoft® Outlook® 2010

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

<1> [Section 2.2.2.1](#): Office Outlook 2007 does not set the **PidTagIconIndex** property ([\[MS-OXPROPS\]](#) section 2.793) on SMS or MMS objects.

<2> [Section 2.2.2.3](#): Office Outlook 2007 SP1 sets both the **PidTagBody** property ([\[MS-OXPROPS\]](#) section 2.676) and the **PidTagHtml** property ([\[MS-OXPROPS\]](#) section 2.789) on SMS objects.

<3> [Section 2.2.2.3](#): Office Outlook 2007 SP1 sets both the **PidTagBody** property ([\[MS-OXPROPS\]](#) section 2.676) and the **PidTagHtml** property ([\[MS-OXPROPS\]](#) section 2.789) on MMS objects.

7 Change Tracking

This section identifies changes that were made to the [MS-OXOSMMS] protocol document between the August 2011 and October 2011 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.
- An extensive rewrite, addition, or deletion of major portions of content.
- The removal of a document from the documentation set.
- Changes made for template compliance.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the language and formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical or language changes were introduced. The technical content of the document is identical to the last released version, but minor editorial and formatting changes, as well as updates to the header and footer information, and to the revision summary, may have been made.

Major and minor changes can be described further using the following change types:

- New content added.
- Content updated.
- Content removed.
- New product behavior note added.
- Product behavior note updated.
- Product behavior note removed.
- New protocol syntax added.
- Protocol syntax updated.
- Protocol syntax removed.
- New content added due to protocol revision.
- Content updated due to protocol revision.
- Content removed due to protocol revision.
- New protocol syntax added due to protocol revision.

- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.
- New content added for template compliance.
- Content updated for template compliance.
- Content removed for template compliance.
- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated**.

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.
- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact protocol@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
1 Introduction	Added information about which sections of the specification are normative and can contain RFC 2119 language.	Y	Content updated for template compliance.
	Removed the section Folders.	N	Content removed.

8 Index

A

[Additional Property Constraints message](#) 9
[Applicability](#) 7

C

[Capability negotiation](#) 7
[Change tracking](#) 19

F

[Fields - vendor-extensible](#) 7

G

[Glossary](#) 5

I

[Implementer - security considerations](#) 17
[Index of security parameters](#) 17
[Informative references](#) 6
[Introduction](#) 5

M

Messages

[Additional Property Constraints](#) 9
[transport](#) 8

N

[Normative references](#) 5

O

[Overview \(synopsis\)](#) 6

P

[Parameters - security index](#) 17
[Preconditions](#) 6
[Prerequisites](#) 6
[Product behavior](#) 18

R

References

[informative](#) 6
[normative](#) 5
[Relationship to other protocols](#) 6

S

Security

[implementer considerations](#) 17
[parameter index](#) 17

[Standards assignments](#) 7

T

[Tracking changes](#) 19
[Transport](#) 8

V

[Vendor-extensible fields](#) 7
[Versioning](#) 7