[MS-OXWSMSG]:
E-Mail Message Types Web Service Protocol Specification

Revision Summary

Date	Revision History	Revision Class	Comments
07/15/2009	1.0	Major	Initial Availability.
11/04/2009	2.0.0	Major	Updated and revised the technical content.
02/10/2010	3.0.0	Major	Updated and revised the technical content.
05/05/2010	3.1.0	Minor	Updated the technical content.
08/04/2010	4.0	Major	Significantly changed the technical content.
11/03/2010	5.0	Major	Significantly changed the technical content.
03/18/2011	6.0	Major	Significantly changed the technical content.
08/05/2011	6.1	Minor	Clarified the meaning of the technical content.
10/07/2011	6.1	No change	No changes to the meaning, language, or formatting of the technical content.
01/20/2012	7.0	Major	Significantly changed the technical content.
04/27/2012	7.0	No change	No changes to the meaning, language, or formatting of the technical content.
07/16/2012	7.0	No change	No changes to the meaning, language, or formatting of the technical content.
10/08/2012	7.1	Minor	Clarified the meaning of the technical content.

Table of Contents

1   Introduction

The E-Mail Message Types Web Service Protocol is used to create, get, update, delete, move, copy, and send e-mail message in a mailbox.

Sections 1.8, 2, and 3 of this specification are normative and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in RFC 2119. Sections 1.5 and 1.9 are also normative but cannot contain those terms. All other sections and examples in this specification are informative.

1.1   Glossary

The following terms are defined in [MS-GLOS]:

Hypertext Transfer Protocol (HTTP)
Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)
XML

The following terms are defined in [MS-OXGLOS]:

endpoint
Junk Email folder
mailbox
Multipurpose Internet Mail Extensions (MIME)
Uniform Resource Locator (URL)
Web Services Description Language (WSDL)
WSDL message
WSDL port type
XML namespace
XML schema

The following terms are specific to this document:

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

References to Microsoft Open Specifications documentation do not include a publishing year because links are to the latest version of the technical documents, which are updated frequently. References to other documents include a publishing year when one is available.

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-OXDSCLI] Microsoft Corporation, "Autodiscover Publishing and Lookup Protocol Specification".

[MS-OXWSADISC] Microsoft Corporation, "Autodiscover Publishing and Lookup SOAP-Based Web Service Protocol Specification".

[MS-OXWSCDATA] Microsoft Corporation, "Common Web Service Data Types".

[MS-OXWSCORE] Microsoft Corporation, "Core Items Web Service Protocol Specification".

[MS-OXWSFOLD] Microsoft Corporation, "Folders and Folder Permissions Web Service Protocol Specification".

[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

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000, http://www.ietf.org/rfc/rfc2818.txt

[SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", May 2000, http://www.w3.org/TR/2000/NOTE-SOAP-20000508/

[WSDL] Christensen, E., Curbera, F., Meredith, G., and Weerawarana, S., "Web Services Description Language (WSDL) 1.1", W3C Note, March 2001, http://www.w3.org/TR/2001/NOTE-wsdl-20010315

[XMLNS] Bray, T., Hollander, D., Layman, A., et al., Eds., "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation, December 2009, http://www.w3.org/TR/2009/REC-xml-names-20091208/

[XMLSCHEMA1] Thompson, H.S., Ed., Beech, D., Ed., Maloney, M., Ed., and Mendelsohn, N., Ed., "XML Schema Part 1: Structures", W3C Recommendation, May 2001, http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/

[XMLSCHEMA2] Biron, P.V., Ed. and Malhotra, A., Ed., "XML Schema Part 2: Datatypes", W3C Recommendation, May 2001, http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/

1.2.2   Informative References

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

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

[MS-OXPROTO] Microsoft Corporation, "Exchange Server Protocols System Overview".

[RFC2616] Fielding, R., Gettys, J., Mogul, J., et al., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999, http://www.ietf.org/rfc/rfc2616.txt

1.3   Overview

The E-Mail Message Types Web Service Protocol provides clients with the ability to create, update, and delete e-mail items on the server. Clients create e-mail items by using the **CreateItem** operation, or they get the properties of an existing task item by using the **GetItem** operation. E-mails can also be sent, updated, deleted, moved, or copied on the server by using the **SendItem**, **UpdateItem**, **DeleteItem**, **MoveItem**, and **CopyItem** operations, respectively.

1.4   Relationship to Other Protocols

A client that implements this protocol can use the Autodiscover Publishing and Lookup SOAP-Based Web Service Protocol, as described in [MS-OXWSADISC], or the Autodiscover Publishing and Lookup Protocol, as described in [MS-OXDSCLI], to identify the target **endpoint (4)** to use for each operation.

This protocol uses the SOAP Protocol, as described in [SOAP1.1], to specify the structure information exchanged between the client and server. This protocol uses the **XML** Protocol, as

described in [XMLSCHEMA1] and [XMLSCHEMA2], to describe the message content sent to and from the server.

This protocol uses SOAP over **HTTP**, as described in [RFC2616], and SOAP over **HTTPS**, as described in [RFC2818], as shown in the following figure.
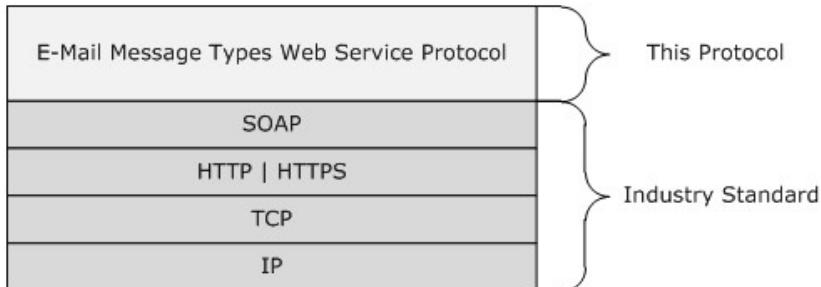
Figure 1: This protocol in relation to other protocols

For conceptual background information and overviews of the relationships and interactions between this and other protocols, see [MS-OXPROTO].

1.5   Prerequisites/Preconditions

The endpoint (4) **URL** that is returned by either the Autodiscover Publishing Lookup SOAP-Based Web Service Protocol, as described in [MS-OXWSADISC], or the Autodiscover Publishing and Lookup Protocol, as described in [MS-OXDSCLI], is required to form the HTTP request to the Web server that hosts this protocol. The operations that this protocol defines cannot be accessed unless the correct endpoint (4) is identified in the HTTP Web requests that target this protocol.

1.6   Applicability Statement

This protocol is applicable to environments that copy, create, delete, get, send, or update e-mails by using Exchange Web Services.

1.7   Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

▪ **Supported Transports:** This protocol uses multiple transports with SOAP 1.1, as specified in section 2.1.

▪ **Protocol Versions:** This protocol has only one **WSDL port type** version. The **WSDL** version of the request is identified by using the **t:RequestServerVersion** element, as described in [MS-OXWSCDATA] section 2.2.5.9, and the version of the server responding to the request is identified by using the **t:ServerVersionInfo** element, as described in [MS-OXWSCDATA] section 2.2.5.10.

▪ **Security and Authentication Methods:** This protocol relies on the Web server that is hosting it to perform authentication.

▪ **Localization:** This protocol includes text strings in various messages. Localization considerations for such strings are specified in section 3.1.4.

▪ **Capability Negotiation:** This protocol does not support version negotiation.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

In the following sections, the schema definition might differ from the processing rules imposed by the protocol. The WSDL in this specification provides a base description of the protocol. The schema in this specification provides a base description of the message syntax. The text that specifies the WSDL and schema might specify restrictions that reflect actual protocol behavior. For example, the schema definition might allow for an element to be **empty**, **null**, or **not present** but the behavior of the protocol as specified restricts the same elements to being **non-empty**, **not null**, or **present**.

2.1 Transport

The SOAP version supported is SOAP 1.1. For details, see [SOAP1.1].

This protocol relies on the Web server that hosts the application to perform authentication. The protocol SHOULD use secure communications via HTTPS, as defined in [RFC2818].

2.2 Common Message Syntax

This section contains common definitions that are used by this protocol. The syntax of the definitions uses **XML schema**, as specified in [XMLSCHEMA1] and [XMLSCHEMA2], and Web Services Description Language (WSDL), as specified in [WSDL].

2.2.1 Namespaces

This specification defines and references various **XML namespaces** by using the mechanisms specified in [XMLNS]. Although this specification associates a specific XML namespace prefix for each XML namespace that is used, the choice of any particular XML namespaces prefix is implementation-specific and is not significant for interoperability.

Prefix	Namespace URI	Reference
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
tns	http://schemas.microsoft.com/exchange/services/2006/messages	
s	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1]
targetNamespace	http://schemas.microsoft.com/exchange/services/2006/messages	
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
t	http://schemas.microsoft.com/exchange/services/2006/types	

2.2.2 Message Syntax

This specification does not define any common **WSDL message** definitions.

2.2.3 Elements

This specification does not define any common XML schema element definitions.

2.2.4 Complex Types

The following table summarizes the set of common XML schema complex type definitions that are defined by this specification. XML schema complex type definitions that are specific to a particular operation are defined with the operation.

Complex type name	Description
MessageType	Represents a server e-mail message in a user's **mailbox**.

2.2.4.1 t:MessageType Complex Type

The **MessageType** complex type represents a server e-mail message in a mailbox. The **MessageType** complex type extends the **ItemType** complex type ([MS-OXWSCORE] section 2.2.4.8).

```
<xs:complexType name="MessageType">
  <xs:complexContent>
    <xs:extension
      base="t:ItemType"
    >
      <xs:sequence>
        <xs:element name="Sender"
          type="t:SingleRecipientType"
          minOccurs="0"
         />
        <xs:element name="ToRecipients"
          type="t:ArrayOfRecipientsType"
          minOccurs="0"
         />
        <xs:element name="CcRecipients"
          type="t:ArrayOfRecipientsType"
          minOccurs="0"
         />
        <xs:element name="BccRecipients"
          type="t:ArrayOfRecipientsType"
          minOccurs="0"
         />
        <xs:element name="IsReadReceiptRequested"
          type="xs:boolean"
          minOccurs="0"
         />
        <xs:element name="IsDeliveryReceiptRequested"
          type="xs:boolean"
          minOccurs="0"
         />
        <xs:element name="ConversationIndex"
          type="xs:base64Binary"
          minOccurs="0"
         />
        <xs:element name="ConversationTopic"
          type="xs:string"
          minOccurs="0"
         />
        <xs:element name="From"
          type="t:SingleRecipientType"
          minOccurs="0"
```

```
    />
  <xs:element name="InternetMessageId"
    type="xs:string"
    minOccurs="0"
    />
  <xs:element name="IsRead"
    type="xs:boolean"
    minOccurs="0"
    />
  <xs:element name="IsResponseRequested"
    type="xs:boolean"
    minOccurs="0"
    />
  <xs:element name="References"
    type="xs:string"
    minOccurs="0"
    />
  <xs:element name="ReplyTo"
    type="t:ArrayOfRecipientsType"
    minOccurs="0"
    />
  <xs:element name="ReceivedBy"
    type="t:SingleRecipientType"
    minOccurs="0"
    />
  <xs:element name="ReceivedRepresenting"
    type="t:SingleRecipientType"
    minOccurs="0"
    />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The following table lists the child elements of the **MessageType** complex type.

Element name	Type	Description
Sender	**t:SingleRecipientType** ([MS-OXWSCDATA] section 2.2.4.60)	Specifies the sender of a message. This element is optional. This is a read/write element.
ToRecipients	**t:ArrayOfRecipientsType** ([MS-OXWSCDATA] section 2.2.4.9)	Specifies a collection of recipients of an e-mail. This element is required for sending a message. This is a read/write element.
CcRecipients	**t:ArrayOfRecipientsType**	Specifies a collection of recipients that receive a carbon copy (Cc) of an e-mail. This element is optional. This is a read/write element.
BccRecipients	**t:ArrayOfRecipientsType**	Specifies a collection of recipients that receive a blind carbon copy (Bcc) of an e-mail. This element is optional. This is a read/write element.

Element name	Type	Description
IsReadReceiptRequested	**xs:boolean** ([XMLSCHEMA2])	Specifies a Boolean value that indicates whether the sender of a message requests a read receipt. This element is optional. This is a read/write element. A text value of "true" indicates that a read receipt is requested from the recipient of the message.
IsDeliveryReceiptRequested	**xs:boolean**	Specifies a Boolean value that indicates whether the sender of the message has requesteda delivery receipt. This element is required when the **IsDeliveryReceiptRequested** element is specified. This is a read/write element. A text value of "true" indicates that a delivery receipt has been requested from the recipient of the message.
ConversationIndex	**xs:base64Binary** ([XMLSCHEMA2])	Specifies the identifier to the thread to which the message belongs. This element is optional. This is a read/write element.
ConversationTopic	**xs:string** ([XMLSCHEMA2])	Specifies the conversation identifier. This element is optional. This is a read/write element.
From	**t:SingleRecipientType**	Specifies the addressee from whom the message was sent. This element is optional. This is a read/write element.
InternetMessageId	**xs:string**	Specifies the Internet message identifier for the message. This element is optional. This is a read/write element.
IsRead	**xs:boolean**	Specifies a Boolean value that indicates whether the message has been read. This element is read-only. The text value of "true" indicates that the message has been read.
IsResponseRequested	**xs:boolean**	Specifies a Boolean value that indicates whether a response to an e-mail has been requested. This element is optional. This is a read/write element. A text value of "true" indicates that a response has been requested.
References	**xs:string**	Specifies the Usenet header that is used to correlate replies with their original message. This element is optional. This is a read/write element.
ReplyTo	**t:ArrayOfRecipientsType**	Specifies a collection of addresses to which replies should be sent. This element is optional. This is a read/write element.

Element name	Type	Description
ReceivedBy	**t:SingleRecipientType**	Identifies the delegate in a delegate access scenario.
ReceivedRepresenting	**t:SingleRecipientType**	Identifies the principal in a delegate access scenario.

2.2.5 Simple Types

The following table summarizes the set of common XML schema simple type definitions that are defined by this specification. XML schema simple type definitions that are specific to a particular operation are described with the operation.

Simple type name	Description
MessageDispositionType	Specifies how a message item is handled after it is created or updated.

2.2.5.1 t:MessageDispositionType Simple Type

The **MessageDispositionType** simple type specifies how a message item is handled after it is created or updated.

```
<xs:simpleType name="MessageDispositionType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="SaveOnly"
     />
    <xs:enumeration
      value="SendOnly"
     />
    <xs:enumeration
      value="SendAndSaveCopy"
     />
  </xs:restriction>
</xs:simpleType>
```

The following table lists the values that are defined by the **MessageDispositionType** simple type.

Value	Meaning
SaveOnly	When used in the **CreateItemType** element ([MS-OXWSCORE] section 3.1.4.2.2.1), the e-mail message item is saved in the folder that is specified by the **TargetFolderIdType** complex type ([MS-OXWSFOLD] section 2.2.4.14). Messages can be sent later by using the **SendItem** operation (section 3.1.4.7) on an **ExchangeServiceBinding** object. In this case, an item identifier is returned.
SendOnly	When used in the **CreateItemType** element ([MS-OXWSCORE] section 3.1.4.2.2.1), the e-mail message item is sent but no copy is saved. In this case, an item identifier is not returned.
SendAndSaveCopy	When used in the **CreateItemType** element ([MS-OXWSCORE] section 3.1.4.2.2.1), the e-mail message item is sent and a copy is saved in the **TargetFolderIdType**

Value	Meaning
	complex type ([MS-OXWSFOLD] section 2.2.4.14). In this case, an item identifier is not returned.

2.2.6 Attributes

This specification does not define any common XML schema attribute definitions.

2.2.7 Groups

This specification does not define any common XML schema group definitions.

2.2.8 Attribute Groups

This specification does not define any common XML schema attribute group definitions.

3   Protocol Details

The client side of this protocol is simply a pass-through. That is, no additional timers or other state is required on the client side of this protocol. Calls made by the higher-layer protocol or application are passed directly to the transport, and the results returned by the transport are passed directly back to the higher-layer protocol or application.

3.1   ExchangeServicePortType Server Details

The E-Mail Message Types Items Web Service Protocol defines a single port type with seven operations. The operations enable client implementations to get, create, delete, update, move, copy, and send messages in a user's mailbox.

3.1.1   Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that specified in this document.

3.1.2   Timers

None.

3.1.3   Initialization

None.

3.1.4   Message Processing Events and Sequencing Rules

The following table summarizes the list of WSDL operations as defined by this specification.

Operation name	Description
CopyItem	Copies a message on the server.
CreateItem	Creates a message on the server.
DeleteItem	Deletes a message from the server.
GetItem	Gets a message from the server.
MoveItem	Moves a message on the server.
SendItem	Sends a message to the server.
UpdateItem	Updates a message on the server.

3.1.4.1   CreateItem

The **CreateItem** operation creates e-mail messages.

The following is the WSDL port type specification of the **CreateItem** operation.

```
<wsdl:operation name="CreateItem">
    <wsdl:input message="tns:CreateItemSoapIn" />
    <wsdl:output message="tns:CreateItemSoapOut" />
</wsdl:operation>
```

The following is the WSDL binding specification of the **CreateItem** operation.

```
<wsdl:operation name="CreateItem">
    <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/CreateItem" />
    <wsdl:input>
        <soap:header message="tns:CreateItemSoapIn" part="Impersonation" use="literal"/>
        <soap:header message="tns:CreateItemSoapIn" part="MailboxCulture" use="literal"/>
        <soap:header message="tns:CreateItemSoapIn" part="RequestVersion" use="literal"/>
        <soap:header message="tns:CreateItemSoapIn" part="TimeZoneContext" use="literal"/>
        <soap:body parts="request" use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body parts="CreateItemResult" use="literal" />
        <soap:header message="tns:CreateItemSoapOut" part="ServerVersion" use="literal"/>
    </wsdl:output>
</wsdl:operation>
```

The protocol client sends a **CreateItemSoapIn** request WSDL message, and the protocol server responds with a **CreateItemSoapOut** response WSDL message.

A successful **CreateItem** operation request returns a **CreateItemResponse** element with the **ResponseClass** attribute of the **CreateItemResponseMessage** element set to "Success". The **ResponseCode** element of the **CreateItemResponse** element is set to "NoError".

For more information, see **CreateItem** as described in [MS-OXWSCORE] section 3.1.4.2.

3.1.4.2   GetItem

The **GetItem** operation enables the user to get e-mail messages and to access information about e-mail messages.

The following is the WSDL port type specification of the **GetItem** operation.

```
<wsdl:operation name="GetItem">
    <wsdl:input message="tns:GetItemSoapIn" />
    <wsdl:output message="tns:GetItemSoapOut" />
</wsdl:operation>
```

The following is the WSDL binding specification of the **GetItem** operation.

```
<wsdl:operation name="GetItem">
    <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/GetItem" />
    <wsdl:input>
        <soap:header message="tns:GetItemSoapIn" part="Impersonation" use="literal"/>
        <soap:header message="tns:GetItemSoapIn" part="MailboxCulture" use="literal"/>
        <soap:header message="tns:GetItemSoapIn" part="RequestVersion" use="literal"/>
        <soap:header message="tns:GetItemSoapIn" part="TimeZoneContext" use="literal"/>
        <soap:body parts="request" use="literal" />
```

```
        </wsdl:input>
        <wsdl:output>
            <soap:body parts="GetItemResult" use="literal" />
            <soap:header message="tns:GetItemSoapOut" part="ServerVersion" use="literal"/>
        </wsdl:output>
    </wsdl:operation>
```

The **GetItem** operation request MUST include the elements listed in the following table.

Element name	Description
ItemShape	Specifies a set of properties to be returned. The child elements for this element are listed in the following table.
ItemIds	Contains the unique identities of items. This element includes the **Id** attribute, which identifies a specific item in the store, and the **ChangeKey** attribute, which identifies a specific version of an item.

The child elements of the **ItemShape** element are listed in the following table.

Element name	Description
BaseShape	Identifies the basic configuration of properties to be returned in an item response. Set this element to **IdOnly** to return only the item ID, or set it to **AllProperties** to return all of the properties used by the server to construct a message. This element MUST be present.
IncludeMimeContent	Indicates whether **MIME** content is included in a returned message or attachment.
BodyType	Indicates whether a message body is returned as HTML.
FilterHtmlContent	Indicates whether to filter unsafe HTML content from a message or attachment.
AdditionalProperties	Identifies additional item properties to be returned in a response.

The protocol client sends a **GetItemSoapIn** request WSDL message, and the protocol server responds with a **GetItemSoapOut** response WSDL message.

A successful **GetItem** operation request returns a **GetItemResponse** element with the **ResponseClass** attribute of the **GetItemResponseMessage** element set to "Success". The **ResponseCode** element of the **GetItemResponse** element is set to "NoError".

For more information, see **Gettem** as described in [MS-OXWSCORE] section 3.1.4.2.

3.1.4.3   UpdateItem

The **UpdateItem** operation updates e-mail message properties in the server store.

The following is the WSDL port type specification of the operation.

```
<wsdl:operation name="UpdateItem">
    <wsdl:input message="tns:UpdateItemSoapIn" />
    <wsdl:output message="tns:UpdateItemSoapOut" />
```

```
    </wsdl:operation>
```

The following is the WSDL binding specification of the **UpdateItem** operation.

```
<wsdl:operation name="UpdateItem">
    <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/UpdateItem" />
    <wsdl:input>
        <soap:header message="tns:UpdateItemSoapIn" part="Impersonation" use="literal"/>
        <soap:header message="tns:UpdateItemSoapIn" part="MailboxCulture" use="literal"/>
        <soap:header message="tns:UpdateItemSoapIn" part="RequestVersion" use="literal"/>
        <soap:header message="tns:UpdateItemSoapIn" part="TimeZoneContext" use="literal"/>
        <soap:body parts="request" use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body parts="UpdateItemResult" use="literal" />
        <soap:header message="tns:UpdateItemSoapOut" part="ServerVersion" use="literal"/>
    </wsdl:output>
</wsdl:operation>
```

The **UpdateItem** operation modifies a message that has already been created and sent. The **UpdateItem** operation request MAY contain the **MessageDispositionType** simple type (section 2.2.5.1) and MUST contain the **ConflictResolutionType** simple type ([MS-OXWSCORE] section 3.1.4.9.4.1).

The protocol client sends an **UpdateItemSoapIn** request WSDL message, and the protocol server responds with an **UpdateItemSoapOut** response WSDL message.

A successful **UpdateItem** operation request returns an **UpdateItemResponse** element with the **ResponseClass** attribute of the **UpdateItemResponseMessage** element set to "Success". The **ResponseCode** element of the **UpdateItemResponse** element is set to "NoError".

For more information, see **UpdateItem** as described in [MS-OXWSCORE] section 3.1.4.9.

3.1.4.4   DeleteItem

The **DeleteItem** operation deletes e-mail messages from the server store.

The following is the WSDL port type specification of the **DeleteItem** operation.

```
<wsdl:operation name="DeleteItem">
    <wsdl:input message="tns:DeleteItemSoapIn" />
    <wsdl:output message="tns:DeleteItemSoapOut" />
</wsdl:operation>
```

The following is the WSDL binding specification of the **DeleteItem** operation.

```
<wsdl:operation name="DeleteItem">
    <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/DeleteItem" />
    <wsdl:input>
        <soap:header message="tns:DeleteItemSoapIn" part="Impersonation" use="literal"/>
        <soap:header message="tns:DeleteItemSoapIn" part="MailboxCulture" use="literal"/>
        <soap:header message="tns:DeleteItemSoapIn" part="RequestVersion" use="literal"/>
```

```
            <soap:body parts="request" use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body parts="DeleteItemResult" use="literal" />
            <soap:header message="tns:DeleteItemSoapOut" part="ServerVersion" use="literal"/>
        </wsdl:output>
    </wsdl:operation>
```

The protocol client sends a **DeleteItemSoapIn** request WSDL message, and the protocol server responds with a **DeleteItemSoapOut** response WSDL message.

A successful **DeleteItem** operation request returns a **DeleteItemResponse** element with the **ResponseClass** attribute of the **DeleteItemResponseMessage** element set to "Success". The **ResponseCode** element of the **DeleteItemResponse** element is set to "NoError".

For more information, see **DeleteItem** as described in [MS-OXWSCORE] section 3.1.4.2.

3.1.4.5   MoveItem

The **MoveItem** operation moves one or more e-mail messages to a single destination folder.

The following is the WSDL port type specification of the **MoveItem** operation.

```
<wsdl:operation name="MoveItem">
    <wsdl:input message="tns:MoveItemSoapIn" />
    <wsdl:output message="tns:MoveItemSoapOut" />
</wsdl:operation>
```

The following is the WSDL binding specification of the **MoveItem** operation.

```
<wsdl:operation name="MoveItem">
    <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/MoveItem" />
    <wsdl:input>
        <soap:header message="tns:MoveItemSoapIn" part="Impersonation" use="literal"/>
        <soap:header message="tns:MoveItemSoapIn" part="MailboxCulture" use="literal"/>
        <soap:header message="tns:MoveItemSoapIn" part="RequestVersion" use="literal"/>
        <soap:body parts="request" use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body parts="MoveItemResult" use="literal" />
        <soap:header message="tns:MoveItemSoapOut" part="ServerVersion" use="literal"/>
    </wsdl:output>
</wsdl:operation>
```

The protocol client sends a **MoveItemSoapIn** request WSDL message, and the protocol server responds with a **MoveItemSoapOut** response WSDL message.

A successful **MoveItem** operation request returns a **MoveItemResponse** element with the **ResponseClass** attribute of the **MoveItemResponseMessage** element set to "Success". The **ResponseCode** element of the **MoveItemResponse** element is set to "NoError".

For more information, see **MoveItem** as described in [MS-OXWSCORE] section 3.1.4.7.

3.1.4.6 CopyItem

The **CopyItem** operation copies e-mail messages and puts them in a different folder.

The following is the WSDL port type specification of the **CopyItem** operation.

```
<wsdl:operation name="CopyItem">
    <wsdl:input message="tns:CopyItemSoapIn" />
    <wsdl:output message="tns:CopyItemSoapOut" />
</wsdl:operation>
```

The following is the WSDL binding specification of the **CopyItem** operation.

```
<wsdl:operation name="CopyItem">
    <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/CopyItem" />
    <wsdl:input>
        <soap:header message="tns:CopyItemSoapIn" part="Impersonation" use="literal"/>
        <soap:header message="tns:CopyItemSoapIn" part="MailboxCulture" use="literal"/>
        <soap:header message="tns:CopyItemSoapIn" part="RequestVersion" use="literal"/>
        <soap:body parts="request" use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body parts="CopyItemResult" use="literal" />
        <soap:header message="tns:CopyItemSoapOut" part="ServerVersion" use="literal"/>
    </wsdl:output>
</wsdl:operation>
```

The protocol client sends a **CopyItemSoapIn** request WSDL message, and the protocol server responds with a **CopyItemSoapOut** response WSDL message.

A successful **CopyItem** operation request returns a **CopyItemResponse** element with the **ResponseClass** attribute of the **CopyItemResponseMessage** element set to "Success". The **ResponseCode** element of the **CopyItemResponse** element is set to "NoError".

For more information, see **CopyItem** as described in [MS-OXWSCORE] section 3.1.4.1.

3.1.4.7 SendItem

The **SendItem** operation sends e-mail messages that are located in the server store.

The following is the WSDL port type specification of the **SendItem** operation.

```
<wsdl:operation name="SendItem ">
    <wsdl:input message="tns:SendItemSoapIn" />
    <wsdl:output message="tns:SendItemSoapOut" />
</wsdl:operation>
```

The following is the WSDL binding specification of the **SendItem** operation.

```
<wsdl:operation name="SendItem">
    <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/SendItem" />
    <wsdl:input>
        <soap:header message="tns:SendItemSoapIn" part="Impersonation" use="literal"/>
```

```
            <soap:header message="tns:SendItemSoapIn" part="MailboxCulture" use="literal"/>
            <soap:header message="tns:SendItemSoapIn" part="RequestVersion" use="literal"/>
            <soap:body parts="request" use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body parts="SendItemResult" use="literal" />
            <soap:header message="tns:SendItemSoapOut" part="ServerVersion" use="literal"/>
        </wsdl:output>
    </wsdl:operation>
```

The protocol client sends a **SendItemSoapIn** request WSDL message, and the protocol server responds with a **SendItemSoapOut** response WSDL message.

A successful **SendItem** operation request returns a **SendItemResponse** element with the **ResponseClass** attribute of the **SendItemResponseMessage** element set to "Success". The **ResponseCode** element of the **SendItemResponse** element is set to "NoError".

For more information, see **SendItem** as described in [MS-OXWSCORE] section 3.1.4.8.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

4 Protocol Examples

The following examples show the request and response XML for the operations that this protocol uses.

4.1 Create Message Example

The following is an example of a **CreateItem** operation that creates a message. This example creates an e-mail message with only the **Subject** and **Body** properties completed, and then it sends the mail to User1 and User2 at Contoso.com. The e-mail message can be set with many more properties, such as attachments, Bcc recipients, categories, sender, and item class. This example shows how to send an e-mail message and save a copy of the message in the default Sent Items folder by using the **SendAndSaveCopy** method.

The client constructs the request XML and sends it to the server. The newly created message is sent to the server.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
      xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
      xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <t:RequestServerVersion Version="Exchange2010" />
  </soap:Header>
  <soap:Body>
    <m:CreateItem MessageDisposition="SendAndSaveCopy">
      <m:SavedItemFolderId>
        <t:DistinguishedFolderId Id="sentitems" />
      </m:SavedItemFolderId>
      <m:Items>
        <t:Message>
          <t:Subject>Interesting</t:Subject>
          <t:Body BodyType="HTML">The merger is finalized.</t:Body>
          <t:ToRecipients>
            <t:Mailbox>
              <t:EmailAddress>User1@Contoso.com</t:EmailAddress>
            </t:Mailbox>
            <t:Mailbox>
              <t:EmailAddress>User2@Contoso.com</t:EmailAddress>
            </t:Mailbox>
          </t:ToRecipients>
        </t:Message>
      </m:Items>
    </m:CreateItem>
  </soap:Body>
</soap:Envelope>
```

The server constructs the response XML and sends it to the client.

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <h:ServerVersionInfo MajorVersion="14"
        MinorVersion="1"
        MajorBuildNumber="63"
```

```
            MinorBuildNumber="0"
            Version="Exchange2010"
            xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
            xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema" />
      </s:Header>
      <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        <m:CreateItemResponse
   xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
            xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
          <m:ResponseMessages>
            <m:CreateItemResponseMessage ResponseClass="Success">
              <m:ResponseCode>NoError</m:ResponseCode>
              <m:Items />
            </m:CreateItemResponseMessage>
          </m:ResponseMessages>
        </m:CreateItemResponse>
      </s:Body>
   </s:Envelope>
```

4.2 Get Message Example

The following is an example of a GetItem operation that gets a message. This example gets a message from the server store. The message is already identified with its **ItemId** Id and **ChangeKey** attributes.

The client constructs the request XML and sends it to the server. Note that the **ItemId** Id and **ChangeKey** attributes have been shortened to preserve readability.

```
   <?xml version="1.0" encoding="utf-8"?>
   <soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
         xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
         xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
     <soap:Header>
       <t:RequestServerVersion Version="Exchange2010" />
     </soap:Header>
     <soap:Body>
       <m:GetItem>
         <m:ItemShape>
           <t:BaseShape>IdOnly</t:BaseShape>
           <t:AdditionalProperties>
             <t:FieldURI FieldURI="item:Body" />
           </t:AdditionalProperties>
         </m:ItemShape>
         <m:ItemIds>
           <t:ItemId Id="AAMkAGY4YzQw" ChangeKey="CQAAABYAAA " />
         </m:ItemIds>
       </m:GetItem>
     </soap:Body>
   </soap:Envelope>
```

The server constructs the response XML and sends it to the client. Note that the **ItemId** Id and **ChangeKey** attributes have been shortened to preserve readability.

```xml
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <h:ServerVersionInfo MajorVersion="14"
        MinorVersion="1"
        MajorBuildNumber="63"
        MinorBuildNumber="0"
        Version="Exchange2010"
        xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
        xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema" />
  </s:Header>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <m:GetItemResponse
 xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
        xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
      <m:ResponseMessages>
        <m:GetItemResponseMessage ResponseClass="Success">
          <m:ResponseCode>NoError</m:ResponseCode>
          <m:Items>
            <t:Message>
              <t:ItemId Id="AAMkAGY4YzQw" ChangeKey="CQAAABYAAA " />
              <t:Body BodyType="HTML">&lt;meta http-equiv="Content-Type"
content="text/html;
                charset=utf-8"&gt;The merger is finalized.</t:Body>
            </t:Message>
          </m:Items>
        </m:GetItemResponseMessage>
      </m:ResponseMessages>
    </m:GetItemResponse>
  </s:Body>
</s:Envelope>
```

4.3   Update Message Example

The following is an example of an **UpdateItem** operation that updates a message. This example updates the subject of an existing message. The message is already identified with its **ItemId** Id and **ChangeKey** attributes.

The client constructs the request XML and sends it to the server. Note that the **ItemId** Id and **ChangeKey** attributes have been shortened to preserve readability.

```xml
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
    xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
    xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <t:RequestServerVersion Version="Exchange2010" />
  </soap:Header>
  <soap:Body>
    <m:UpdateItem MessageDisposition="SaveOnly" ConflictResolution="AlwaysOverwrite">
      <m:ItemChanges>
        <t:ItemChange>
          <t:ItemId Id="AAMkAGIw" ChangeKey="CQAAABYAAA" />
```

```
            <t:Updates>
              <t:SetItemField>
                <t:FieldURI FieldURI="item:Subject" />
                <t:Message>
                  <t:Subject>Modified and updated mail</t:Subject>
                </t:Message>
              </t:SetItemField>
            </t:Updates>
          </t:ItemChange>
        </m:ItemChanges>
      </m:UpdateItem>
    </soap:Body>
  </soap:Envelope>
```

The server constructs the response XML and sends it to the client. Note that the **ItemId** Id and **ChangeKey** attributes have been shortened to preserve readability.

```
    <?xml version="1.0" encoding="utf-8"?>
    <s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
      <s:Header>
        <h:ServerVersionInfo MajorVersion="14"
            MinorVersion="1"
            MajorBuildNumber="63"
            MinorBuildNumber="0"
            Version="Exchange2010"
            xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
            xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema" />
      </s:Header>
      <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        <m:UpdateItemResponse
  xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
            xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
          <m:ResponseMessages>
            <m:UpdateItemResponseMessage ResponseClass="Success">
              <m:ResponseCode>NoError</m:ResponseCode>
              <m:Items>
                <t:Message>
                  <t:ItemId Id="AAMkAGIwODEy=" ChangeKey="CQAAABYAAA" />
                </t:Message>
              </m:Items>
              <m:ConflictResults>
                <t:Count>0</t:Count>
              </m:ConflictResults>
            </m:UpdateItemResponseMessage>
          </m:ResponseMessages>
        </m:UpdateItemResponse>
      </s:Body>
    </s:Envelope>
```

4.4   Delete Message Example

The following is an example of a **DeleteItem** operation that deletes a message. This example deletes the identified message from the server store.

The client constructs the request XML and sends it to the server. Note that the **ItemId** Id attribute is shortened to preserve readability.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
       xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
       xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <t:RequestServerVersion Version="Exchange2010" />
  </soap:Header>
  <soap:Body>
    <m:DeleteItem DeleteType="HardDelete">
      <m:ItemIds>
        <t:ItemId Id="AAMkAGY4YzQw" />
      </m:ItemIds>
    </m:DeleteItem>
  </soap:Body>
</soap:Envelope>
```

The server constructs the response XML and sends it to the client.

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <h:ServerVersionInfo MajorVersion="14"
        MinorVersion="1"
        MajorBuildNumber="63"
        MinorBuildNumber="0"
        Version="Exchange2010"
        xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
        xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema" />
  </s:Header>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <m:DeleteItemResponse
 xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
        xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
      <m:ResponseMessages>
        <m:DeleteItemResponseMessage ResponseClass="Success">
          <m:ResponseCode>NoError</m:ResponseCode>
        </m:DeleteItemResponseMessage>
      </m:ResponseMessages>
    </m:DeleteItemResponse>
  </s:Body>
</s:Envelope>
```

4.5 Move Message Example

The following is an example of a **MoveItem** operation that moves a message to a specific folder. This example moves an identified message to the **Junk E-mail folder**.

The client constructs the request XML and sends it to the server. Note that the **ItemId** Id attribute has been shortened to preserve readability.

```xml
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
        xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
        xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <t:RequestServerVersion Version="Exchange2010" />
  </soap:Header>
  <soap:Body>
    <m:MoveItem>
      <m:ToFolderId>
        <t:DistinguishedFolderId Id="junkemail" />
      </m:ToFolderId>
      <m:ItemIds>
        <t:ItemId Id="AAMkAGIw " />
      </m:ItemIds>
    </m:MoveItem>
  </soap:Body>
</soap:Envelope>
```

The server constructs the response XML and sends it to the client. Note that the **ItemId** Id and **ChangeKey** attributes have been shortened to preserve readability.

```xml
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <h:ServerVersionInfo MajorVersion="14"
        MinorVersion="1"
        MajorBuildNumber="63"
        MinorBuildNumber="0"
        Version="Exchange2010"
        xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
        xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema" />
  </s:Header>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <m:MoveItemResponse
xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
        xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
      <m:ResponseMessages>
        <m:MoveItemResponseMessage ResponseClass="Success">
          <m:ResponseCode>NoError</m:ResponseCode>
          <m:Items>
            <t:Message>
              <t:ItemId Id="AAMkAGIw" ChangeKey="CQAAABYAAA" />
            </t:Message>
          </m:Items>
        </m:MoveItemResponseMessage>
      </m:ResponseMessages>
    </m:MoveItemResponse>
  </s:Body>
</s:Envelope>
```

4.6 Copy Message Example

The following is an example of a **CopyItem** operation that copies a message to another folder. This example copies an identified message to the Junk E-mail folder.

The client constructs the request XML and sends it to the server. Note that the **ItemId** Id and **ChangeKey** attributes have been shortened to preserve readability.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
      xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
      xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <t:RequestServerVersion Version="Exchange2010" />
  </soap:Header>
  <soap:Body>
    <m:CopyItem>
      <m:ToFolderId>
        <t:DistinguishedFolderId Id="junkemail" />
      </m:ToFolderId>
      <m:ItemIds>
        <t:ItemId Id="AAMkAGIw" ChangeKey="CQAAABYAAA" />
      </m:ItemIds>
    </m:CopyItem>
  </soap:Body>
</soap:Envelope>
```

The server constructs the response XML and sends it to the client. Note that the **ItemId** Id and **ChangeKey** attributes have been shortened to preserve readability.

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <h:ServerVersionInfo MajorVersion="14"
        MinorVersion="1"
        MajorBuildNumber="63"
        MinorBuildNumber="0"
        Version="Exchange2010"
        xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
        xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" />
  </s:Header>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <m:CopyItemResponse
xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
        xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
      <m:ResponseMessages>
        <m:CopyItemResponseMessage ResponseClass="Success">
          <m:ResponseCode>NoError</m:ResponseCode>
          <m:Items>
            <t:Message>
              <t:ItemId Id="AAMkAGIw" ChangeKey="CQAAABYAAA" />
            </t:Message>
          </m:Items>
```

```
            </m:CopyItemResponseMessage>
          </m:ResponseMessages>
        </m:CopyItemResponse>
      </s:Body>
    </s:Envelope>
```

4.7   Send Message Example

The following is an example of a **SendItem** operation that sends a message to the server. This example sends an identified message.

The client constructs the request XML and sends it to the server. Note that the **ItemId** Id and **ChangeKey** attributes have been shortened to preserve readability.

```
    <?xml version="1.0" encoding="utf-8"?>
    <soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
          xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
          xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
      <soap:Header>
        <t:RequestServerVersion Version="Exchange2010" />
      </soap:Header>
      <soap:Body>
        <m:SendItem SaveItemToFolder="false">
          <m:ItemIds>
            <t:ItemId Id="AAMkAGIw" ChangeKey="CQAAABYAAA" />
          </m:ItemIds>
        </m:SendItem>
      </soap:Body>
    </soap:Envelope>
  </EwsLogEntry>
```

The server constructs the response XML and sends it to the client.

```
    <?xml version="1.0" encoding="utf-8"?>
    <s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
      <s:Header>
        <h:ServerVersionInfo MajorVersion="14"
            MinorVersion="1"
            MajorBuildNumber="63"
            MinorBuildNumber="0"
            Version="Exchange2010"
            xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
            xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" />
      </s:Header>
      <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        <m:SendItemResponse
  xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
            xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
          <m:ResponseMessages>
            <m:SendItemResponseMessage ResponseClass="Success">
              <m:ResponseCode>NoError</m:ResponseCode>
            </m:SendItemResponseMessage>
```

```
            </m:ResponseMessages>
          </m:SendItemResponse>
        </s:Body>
      </s:Envelope>
    </EwsLogEntry>
```

5 Security

5.1 Security Considerations for Implementers

None.

5.2 Index of Security Parameters

None.

6   Appendix A: Full WSDL

The following table lists the XML files that are required to implement the functionality that is specified in this document.

File name	Description	Section
MS-OXWSMSG.wsdl	Contains the WSDL for the implementation of this protocol.	6
MS-OXWSMSG-types.xsd	Contains the XML schema type definitions used in this protocol.	7.1

These files have to be placed in a common folder for the WSDL to validate and operate. Also, any schema files that are included in or imported into the MS-OXWSMSG-types.xsd schema or the MS-OXWSMSG-messages.xsd schema have to be placed in the common folder with these files.

This section contains the contents of the MS-OXWSMSG.wsdl file.

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:s="http://www.w3.org/2001/XMLSchema" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages">
  <wsdl:types>
    <xs:schema id="messages" elementFormDefault="qualified" version="Exchange2013"
xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
      <xs:import namespace="http://schemas.microsoft.com/exchange/services/2006/types"/>
      <xs:include schemaLocation="MS-OXWSCORE-messages.xsd"/>
      <!-- Add global elements and types from messages.xsd -->
    </xs:schema>
    <xs:schema id="types" elementFormDefault="qualified" version="Exchange2013"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
      <!-- Add global elements and types from types.xsd -->
    </xs:schema>
  </wsdl:types>
  <wsdl:message name="CopyItemSoapIn">
    <wsdl:part name="request" element="tns:CopyItem"/>
    <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
    <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
    <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
  </wsdl:message>
  <wsdl:message name="CopyItemSoapOut">
    <wsdl:part name="CopyItemResult" element="tns:CopyItemResponse"/>
    <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
  </wsdl:message>
  <wsdl:message name="CreateItemSoapIn">
    <wsdl:part name="request" element="tns:CreateItem"/>
    <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
    <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
```

```
      <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
      <wsdl:part name="TimeZoneContext" element="t:TimeZoneContext"/>
    </wsdl:message>
    <wsdl:message name="CreateItemSoapOut">
      <wsdl:part name="CreateItemResult" element="tns:CreateItemResponse"/>
      <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
    </wsdl:message>
    <wsdl:message name="DeleteItemSoapIn">
      <wsdl:part name="request" element="tns:DeleteItem"/>
      <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
      <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
      <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
    </wsdl:message>
    <wsdl:message name="DeleteItemSoapOut">
      <wsdl:part name="DeleteItemResult" element="tns:DeleteItemResponse"/>
      <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
    </wsdl:message>
    <wsdl:message name="GetItemSoapIn">
      <wsdl:part name="request" element="tns:GetItem"/>
      <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
      <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
      <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
      <wsdl:part name="TimeZoneContext" element="t:TimeZoneContext"/>
    </wsdl:message>
    <wsdl:message name="GetItemSoapOut">
      <wsdl:part name="GetItemResult" element="tns:GetItemResponse"/>
      <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
    </wsdl:message>
    <wsdl:message name="MoveItemSoapIn">
      <wsdl:part name="request" element="tns:MoveItem"/>
      <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
      <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
      <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
    </wsdl:message>
    <wsdl:message name="MoveItemSoapOut">
      <wsdl:part name="MoveItemResult" element="tns:MoveItemResponse"/>
      <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
    </wsdl:message>
    <wsdl:message name="SendItemSoapIn">
      <wsdl:part name="request" element="tns:SendItem"/>
      <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
      <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
      <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
    </wsdl:message>
    <wsdl:message name="SendItemSoapOut">
      <wsdl:part name="SendItemResult" element="tns:SendItemResponse"/>
      <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
    </wsdl:message>
    <wsdl:message name="UpdateItemSoapIn">
      <wsdl:part name="request" element="tns:UpdateItem"/>
      <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
      <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
      <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
      <wsdl:part name="TimeZoneContext" element="t:TimeZoneContext"/>
    </wsdl:message>
    <wsdl:message name="UpdateItemSoapOut">
      <wsdl:part name="UpdateItemResult" element="tns:UpdateItemResponse"/>
      <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
    </wsdl:message>
```

```
<wsdl:portType name="ExchangeServicePortType">
  <wsdl:operation name="CopyItem">
    <wsdl:input message="tns:CopyItemSoapIn"/>
    <wsdl:output message="tns:CopyItemSoapOut"/>
  </wsdl:operation>
  <wsdl:operation name="CreateItem">
    <wsdl:input message="tns:CreateItemSoapIn"/>
    <wsdl:output message="tns:CreateItemSoapOut"/>
  </wsdl:operation>
  <wsdl:operation name="DeleteItem">
    <wsdl:input message="tns:DeleteItemSoapIn"/>
    <wsdl:output message="tns:DeleteItemSoapOut"/>
  </wsdl:operation>
  <wsdl:operation name="GetItem">
    <wsdl:input message="tns:GetItemSoapIn"/>
    <wsdl:output message="tns:GetItemSoapOut"/>
  </wsdl:operation>
  <wsdl:operation name="MoveItem">
    <wsdl:input message="tns:MoveItemSoapIn"/>
    <wsdl:output message="tns:MoveItemSoapOut"/>
  </wsdl:operation>
  <wsdl:operation name="SendItem">
    <wsdl:input message="tns:SendItemSoapIn"/>
    <wsdl:output message="tns:SendItemSoapOut"/>
  </wsdl:operation>
  <wsdl:operation name="UpdateItem">
    <wsdl:input message="tns:UpdateItemSoapIn"/>
    <wsdl:output message="tns:UpdateItemSoapOut"/>
  </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="ExchangeServiceBinding" type="tns:ExchangeServicePortType">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:documentation>
    <wsi:Claim conformsTo="http://ws-i.org/profiles/basic/1.0" xmlns:wsi="http://ws-
i.org/schemas/conformanceClaim/"/>
  </wsdl:documentation>
  <wsdl:operation name="CopyItem">
    <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/CopyItem"/>
    <wsdl:input>
      <soap:header message="tns:CopyItemSoapIn" part="Impersonation" use="literal"/>
      <soap:header message="tns:CopyItemSoapIn" part="MailboxCulture" use="literal"/>
      <soap:header message="tns:CopyItemSoapIn" part="RequestVersion" use="literal"/>
      <soap:body parts="request" use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body parts="CopyItemResult" use="literal"/>
      <soap:header message="tns:CopyItemSoapOut" part="ServerVersion" use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="CreateItem">
    <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/CreateItem"/>
    <wsdl:input>
      <soap:header message="tns:CreateItemSoapIn" part="Impersonation" use="literal"/>
      <soap:header message="tns:CreateItemSoapIn" part="MailboxCulture" use="literal"/>
      <soap:header message="tns:CreateItemSoapIn" part="RequestVersion" use="literal"/>
      <soap:header message="tns:CreateItemSoapIn" part="TimeZoneContext" use="literal"/>
      <soap:body parts="request" use="literal"/>
```

```
        </wsdl:input>
        <wsdl:output>
          <soap:body parts="CreateItemResult" use="literal"/>
          <soap:header message="tns:CreateItemSoapOut" part="ServerVersion" use="literal"/>
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="DeleteItem">
        <soap:operation
  soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/DeleteItem"/>
        <wsdl:input>
          <soap:header message="tns:DeleteItemSoapIn" part="Impersonation" use="literal"/>
          <soap:header message="tns:DeleteItemSoapIn" part="MailboxCulture" use="literal"/>
          <soap:header message="tns:DeleteItemSoapIn" part="RequestVersion" use="literal"/>
          <soap:body parts="request" use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <soap:body parts="DeleteItemResult" use="literal"/>
          <soap:header message="tns:DeleteItemSoapOut" part="ServerVersion" use="literal"/>
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="GetItem">
        <soap:operation
  soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/GetItem"/>
        <wsdl:input>
          <soap:header message="tns:GetItemSoapIn" part="Impersonation" use="literal"/>
          <soap:header message="tns:GetItemSoapIn" part="MailboxCulture" use="literal"/>
          <soap:header message="tns:GetItemSoapIn" part="RequestVersion" use="literal"/>
          <soap:header message="tns:GetItemSoapIn" part="TimeZoneContext" use="literal"/>
          <soap:body parts="request" use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <soap:body parts="GetItemResult" use="literal"/>
          <soap:header message="tns:GetItemSoapOut" part="ServerVersion" use="literal"/>
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="MoveItem">
        <soap:operation
  soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/MoveItem"/>
        <wsdl:input>
          <soap:header message="tns:MoveItemSoapIn" part="Impersonation" use="literal"/>
          <soap:header message="tns:MoveItemSoapIn" part="MailboxCulture" use="literal"/>
          <soap:header message="tns:MoveItemSoapIn" part="RequestVersion" use="literal"/>
          <soap:body parts="request" use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <soap:body parts="MoveItemResult" use="literal"/>
          <soap:header message="tns:MoveItemSoapOut" part="ServerVersion" use="literal"/>
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="UpdateItem">
        <soap:operation
  soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/UpdateItem"/>
        <wsdl:input>
          <soap:header message="tns:UpdateItemSoapIn" part="Impersonation" use="literal"/>
          <soap:header message="tns:UpdateItemSoapIn" part="MailboxCulture" use="literal"/>
          <soap:header message="tns:UpdateItemSoapIn" part="RequestVersion" use="literal"/>
          <soap:header message="tns:UpdateItemSoapIn" part="TimeZoneContext" use="literal"/>
          <soap:body parts="request" use="literal"/>
        </wsdl:input>
```

```
      <wsdl:output>
        <soap:body parts="UpdateItemResult" use="literal"/>
        <soap:header message="tns:UpdateItemSoapOut" part="ServerVersion" use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SendItem">
      <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/SendItem"/>
      <wsdl:input>
        <soap:header message="tns:SendItemSoapIn" part="Impersonation" use="literal"/>
        <soap:header message="tns:SendItemSoapIn" part="MailboxCulture" use="literal"/>
        <soap:header message="tns:SendItemSoapIn" part="RequestVersion" use="literal"/>
        <soap:body parts="request" use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body parts="SendItemResult" use="literal"/>
        <soap:header message="tns:SendItemSoapOut" part="ServerVersion" use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
</wsdl:definitions>
```

7 Appendix B: Full XML Schema

For ease of implementation, the following sections provide the full XML schema for this protocol.

Schema name	Prefix	Section
Types schema	t:	7.1

These files have to be placed in a common folder in order for the WSDL to validate and operate. Also, any schema files that are included in or imported into the MS-OXWSMSG-types.xsd schema have to be placed in the common folder along with the files listed in the table.

7.1 Types Schema

This section contains the contents of the MS-OXWSMSG-types.xsd file and information about additional files that this schema file requires to operate correctly.

MS-OXWSMSG-types.xsd includes the file listed in the following table. For the schema file to operate correctly, this file has to be present in the folder that contains the WSDL and types schema files for this protocol.

File name	Defining specification
MS-OXWSCORE-types.xsd	[MS-OXWSCORE] section 7.2

```xml
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types"
elementFormDefault="qualified" version="Exchange2013" id="types">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xs:include schemaLocation="MS-OXWSCORE-types.xsd"/>
  <xs:complexType name="MessageType">
    <xs:complexContent>
      <xs:extension base="t:ItemType">
        <xs:sequence>
          <xs:element name="Sender" type="t:SingleRecipientType" minOccurs="0"/>
          <xs:element name="ToRecipients" type="t:ArrayOfRecipientsType" minOccurs="0"/>
          <xs:element name="CcRecipients" type="t:ArrayOfRecipientsType" minOccurs="0"/>
          <xs:element name="BccRecipients" type="t:ArrayOfRecipientsType" minOccurs="0"/>
          <xs:element name="IsReadReceiptRequested" type="xs:boolean" minOccurs="0"/>
          <xs:element name="IsDeliveryReceiptRequested" type="xs:boolean" minOccurs="0"/>
          <xs:element name="ConversationIndex" type="xs:base64Binary" minOccurs="0"/>
          <xs:element name="ConversationTopic" type="xs:string" minOccurs="0"/>
          <xs:element name="From" type="t:SingleRecipientType" minOccurs="0"/>
          <xs:element name="InternetMessageId" type="xs:string" minOccurs="0"/>
          <xs:element name="IsRead" type="xs:boolean" minOccurs="0"/>
          <xs:element name="IsResponseRequested" type="xs:boolean" minOccurs="0"/>
          <xs:element name="References" type="xs:string" minOccurs="0"/>
          <xs:element name="ReplyTo" type="t:ArrayOfRecipientsType" minOccurs="0"/>
          <xs:element name="ReceivedBy" type="t:SingleRecipientType" minOccurs="0"/>
          <xs:element name="ReceivedRepresenting" type="t:SingleRecipientType"
minOccurs="0"/>
        </xs:sequence>
```

```
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>

    <xs:simpleType name="MessageDispositionType">
      <xs:restriction base="xs:string">
        <xs:enumeration value="SaveOnly" />
        <xs:enumeration value="SendOnly" />
        <xs:enumeration value="SendAndSaveCopy" />
      </xs:restriction>
    </xs:simpleType>
  </xs:schema>
```

8 Appendix C: 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 2007

- Microsoft® Exchange Server 2010

- Microsoft® Exchange Server 2013

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.

9 Change Tracking

This section identifies changes that were made to the [MS-OXWSMSG] protocol document between the July 2012 and October 2012 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.

Release: October 8, 2012

- 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
1.2.2 Informative References	Added the reference [MS-OXPROTO].	N	Content updated.
1.4 Relationship to Other Protocols	Added informative reference information for overview of relationships between this and other protocols.	N	Content updated.
6 Appendix A: Full WSDL	Changed the version identifier from Exchange2012 to Exchange2013.	N	Content updated.
7.1 Types Schema	Changed the version identifier from Exchange2012 to Exchange2013.	N	Content updated.

10 Index