

[MS-OXWSPOST]: Post Items Web Service Protocol

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 [Open Specification Promise](#) or the [Community Promise](#). 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. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.
- **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.

Preliminary Documentation. This Open Specification provides documentation for past and current releases and/or for the pre-release version of this technology. This Open Specification is final documentation for past or current releases as specifically noted in the document, as applicable; it is preliminary documentation for the pre-release versions. Microsoft will release final documentation in connection with the commercial release of the updated or new version of this technology. As the documentation may change between this preliminary version and the final version of this technology, there are risks in relying on preliminary documentation. To the extent that you incur additional

development obligations or any other costs as a result of relying on this preliminary documentation, you do so at your own risk.

Preliminary

Revision Summary

Date	Revision History	Revision Class	Comments
7/15/2009	1.0	Major	Initial Availability.
11/4/2009	2.0.0	Major	Updated and revised the technical content.
2/10/2010	2.1.0	Minor	Updated the technical content.
5/5/2010	2.1.1	Editorial	Revised and edited the technical content.
8/4/2010	3.0	Major	Significantly changed the technical content.
11/3/2010	3.0	No change	No changes to the meaning, language, or formatting of the technical content.
3/18/2011	3.0	No change	No changes to the meaning, language, or formatting of the technical content.
8/5/2011	4.0	Major	Significantly changed the technical content.
10/7/2011	4.1	Minor	Clarified the meaning of the technical content.
1/20/2012	5.0	Major	Significantly changed the technical content.
4/27/2012	5.0	No Change	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	5.1	Minor	Clarified the meaning of the technical content.
10/8/2012	5.2	Minor	Clarified the meaning of the technical content.
2/11/2013	5.2	No Change	No changes to the meaning, language, or formatting of the technical content.
7/26/2013	5.3	Minor	Clarified the meaning of the technical content.
11/18/2013	5.3	No Change	No changes to the meaning, language, or formatting of the technical content.
2/10/2014	5.3	No Change	No changes to the meaning, language, or formatting of the technical content.
4/30/2014	5.4	Minor	Clarified the meaning of the technical content.
7/31/2014	5.5	Minor	Clarified the meaning of the technical content.
10/30/2014	5.5	No Change	No changes to the meaning, language, or formatting of the technical content.
5/26/2015	6.0	Major	Significantly changed the technical content.

Table of Contents

1	Introduction	6
1.1	Glossary	6
1.2	References	7
1.2.1	Normative References	7
1.2.2	Informative References	8
1.3	Overview	8
1.4	Relationship to Other Protocols	8
1.5	Prerequisites/Preconditions	9
1.6	Applicability Statement	9
1.7	Versioning and Capability Negotiation	9
1.8	Vendor-Extensible Fields	9
1.9	Standards Assignments.....	10
2	Messages.....	11
2.1	Transport.....	11
2.2	Common Message Syntax	11
2.2.1	Namespaces	11
2.2.2	Messages.....	11
2.2.3	Elements	11
2.2.4	Complex Types.....	12
2.2.4.1	t:PostItemType Complex Type.....	12
2.2.4.2	t:PostReplyItemBaseType Complex Type	13
2.2.4.3	t:PostReplyItemType Complex Type	14
2.2.5	Simple Types	15
2.2.6	Attributes	15
2.2.7	Groups	15
2.2.8	Attribute Groups.....	15
3	Protocol Details	16
3.1	ExchangeServicePortType Server Details.....	16
3.1.1	Abstract Data Model.....	16
3.1.2	Timers	16
3.1.3	Initialization.....	16
3.1.4	Message Processing Events and Sequencing Rules	16
3.1.4.1	CopyItem Operation	16
3.1.4.2	CreateItem Operation.....	17
3.1.4.3	DeleteItem Operation	18
3.1.4.4	GetItem Operation	19
3.1.4.5	MoveItem Operation.....	20
3.1.4.6	UpdateItem Operation	21
3.1.5	Timer Events.....	22
3.1.6	Other Local Events.....	22
4	Protocol Examples	23
4.1	Copying a Post Object.....	23
4.2	Creating a Post Object	23
4.3	Deleting a Post Object	24
4.4	Moving a Post Object.....	25
4.5	Retrieving a Post Object.....	25
4.6	Updating a Post Object	27
5	Security	28
5.1	Security Considerations for Implementers	28
5.2	Index of Security Parameters	28

6	Appendix A: Full WSDL	29
7	Appendix B: Full XML Schema.....	33
7.1	Messages Schema	33
7.2	Types Schema	33
8	Appendix C: Product Behavior	35
9	Change Tracking.....	36
10	Index.....	38

Preliminary

1 Introduction

The Post Items Web Service Protocol enables a client to create, retrieve, update, move, copy, and delete **Post objects** on the server.

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 [\[RFC2119\]](#). Sections 1.5 and 1.9 are also normative but do not contain those terms. All other sections and examples in this specification are informative.

1.1 Glossary

The following terms are specific to this document:

conversation: A single representation of a send/response series of email messages. A conversation appears in the Inbox as one unit and allows the user to view and read the series of related email messages in a single effort.

endpoint: A communication port that is exposed by an application server for a specific shared service and to which messages can be addressed.

header: A name-value pair that supplies structured data in an Internet email message or MIME entity.

Hypertext Transfer Protocol (HTTP): An application-level protocol for distributed, collaborative, hypermedia information systems (text, graphic images, sound, video, and other multimedia files) on the World Wide Web.

Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS): An extension of **HTTP** that securely encrypts and decrypts webpage requests.

message store: A unit of containment for a single hierarchy of Folder objects, such as a mailbox or public folders.

Post object: A Message object that represents an entry in a discussion thread stored in a messaging store.

public folder: A Folder object that is stored in a location that is publicly available.

SOAP: A lightweight protocol for exchanging structured information in a decentralized, distributed environment. **SOAP** uses **XML** technologies to define an extensible messaging framework, which provides a message construct that can be exchanged over a variety of underlying protocols. The framework has been designed to be independent of any particular programming model and other implementation-specific semantics. SOAP 1.2 supersedes SOAP 1.1. See [\[SOAP1.2-1/2003\]](#).

Uniform Resource Locator (URL): A string of characters in a standardized format that identifies a document or resource on the World Wide Web. The format is as specified in [\[RFC1738\]](#).

web server: A server computer that hosts websites and responds to requests from applications.

Web Services Description Language (WSDL): An XML format for describing network services as a set of endpoints that operate on messages that contain either document-oriented or procedure-oriented information. The operations and messages are described abstractly and are bound to a concrete network protocol and message format in order to define an endpoint. Related concrete endpoints are combined into abstract endpoints, which describe a network service. WSDL is extensible, which allows the description of endpoints and their messages regardless of the message formats or network protocols that are used.

WSDL message: An abstract, typed definition of the data that is communicated during a WSDL operation [[WSDL](#)]. Also, an element that describes the data being exchanged between web service providers and clients.

WSDL port type: A named set of logically-related, abstract **Web Services Description Language (WSDL)** operations and messages.

XML: The Extensible Markup Language, as described in [[XML1.0](#)].

XML namespace: A collection of names that is used to identify elements, types, and attributes in XML documents identified in a URI reference [[RFC3986](#)]. A combination of XML namespace and local name allows XML documents to use elements, types, and attributes that have the same names but come from different sources. For more information, see [[XMLNS-2ED](#)].

XML namespace prefix: An abbreviated form of an **XML namespace**, as described in [[XML](#)].

XML schema: A description of a type of XML document that is typically expressed in terms of constraints on the structure and content of documents of that type, in addition to the basic syntax constraints that are imposed by **XML** itself. An XML schema provides a view of a document type at a relatively high level of abstraction.

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

1.2 References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the [Errata](#).

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.

[MS-OXDCLI] Microsoft Corporation, "[Autodiscover Publishing and Lookup Protocol](#)".

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

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

[MS-OXWSCORE] Microsoft Corporation, "[Core Items Web Service Protocol](#)".

[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>

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

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

[RFC2822] Resnick, P., Ed., "Internet Message Format", RFC 2822, April 2001, <http://www.ietf.org/rfc/rfc2822.txt>

[RFC850] Horton, M., "Standard for Interchange of USENET Messages", RFC 850, June 1983, <http://www.rfc-editor.org/rfc/rfc850.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., Beech, D., Maloney, M., and Mendelsohn, N., Eds., "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-OXPROTO] Microsoft Corporation, "[Exchange Server Protocols System Overview](#)".

[MS-OXWSSRCH] Microsoft Corporation, "[Mailbox Search Web Service Protocol](#)".

1.3 Overview

This protocol enables clients to create, retrieve, update, move, copy, and delete Post objects on the server. Clients can use the data types and operations described by this protocol to manage post items.

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** 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 the 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 can use the item identifier(s) returned by the Mailbox Search Web Service Protocol, as described in [\[MS-OXWSSRCH\]](#), to access Post objects on the server.

This protocol uses operations that are described in [\[MS-OXWSCORE\]](#) to retrieve, delete, update, move, copy, and create Post objects on the server. For more information about these operations, see section [3.1.4](#).

This protocol uses SOAP over **HTTP**, as described in [\[RFC2616\]](#), and SOAP over **HTTPS**, as described in [\[RFC2818\]](#), as shown in the following layering diagram.

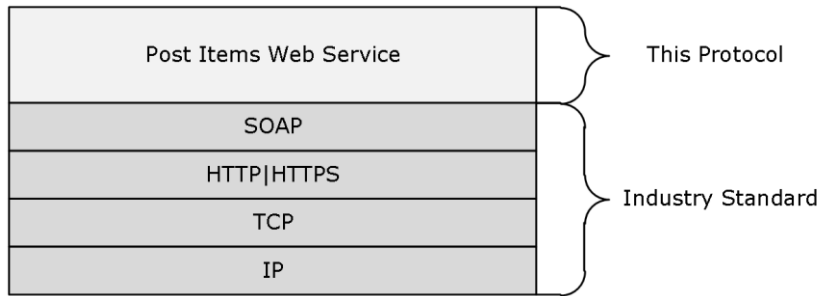


Figure 1: This protocol in relation to other protocols

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 **Uniform Resource Locator (URL)** that is returned by either the Autodiscover Publishing Lookup SOAP-Based Web Service Protocol, as specified in [\[MS-OXWSADISC\]](#), or the Autodiscover Publishing and Lookup Protocol, specified in [\[MS-OXDSCLI\]](#), is required to form the HTTP request to the **Web server** that hosts this protocol. To retrieve the endpoint as described in either the Autodiscover Publishing Lookup SOAP-Based Web Service Protocol or the Autodiscover Publishing and Lookup Protocol, the client needs to have a valid mail-enabled account. The operations that this protocol defines cannot be accessed unless the correct endpoint is identified in the HTTP Web requests that target this protocol.

1.6 Applicability Statement

This protocol is applicable to client programs that create, move, copy, modify, delete, or retrieve Post objects in the server **message store**.

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

- **Supported Transports:** This protocol uses SOAP 1.1, as specified in section [2.1 and in \[SOAP1.1\]](#).
- **Protocol Versions:** This protocol specifies only one **WSDL port type** version. The **WSDL** version of the request is identified by using the RequestServerVersion element, as described in [\[MS-OXWSCDATA\]](#) section 2.2.3.11, and the version of the server responding to the request is identified by using the **ServerVersionInfo** element, as described in [\[MS-OXWSCDATA\]](#) section 2.2.3.12.
- **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.

Preliminary

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

Messages are transported by using SOAP version 1.1, as specified in [\[SOAP1.1\]](#).

This protocol relies on the Web server that hosts the application to perform authentication. The protocol **MUST** support SOAP over HTTP, as specified in [\[RFC2616\]](#), and **SHOULD** support SOAP over HTTPS, as specified 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 defined in [\[XMLSCHEMA1\]](#) and [\[XMLSCHEMA2\]](#), and Web Services Description Language (WSDL), as defined 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** with each XML namespace that is used, the choice of any particular XML namespace 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	[XMLSCHEMA2]
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA2]
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
t	http://schemas.microsoft.com/exchange/services/2006/types	

2.2.2 Messages

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 defined by this specification. XML schema complex type definitions that are specific to a particular operation are described with the operation.

Complex type name	Description
PostItemType	Represents a Post object in a server message store.
PostReplyItemBaseType	Represents the base type for the PostReplyItemType complex type.
PostReplyItemType	Contains a reply to a Post object.

2.2.4.1 t:PostItemType Complex Type

The **PostItemType** complex type represents a Post object in a server message store. The **PostItemType** complex type extends the **ItemType** complex type, as specified in [\[MS-OXWScore\]](#) section 2.2.4.23.

```
<xs:complexType name="PostItemType">
  <xs:complexContent>
    <xs:extension
      base="t:ItemType"
    >
      <xs:sequence>
        <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="PostedTime"
          type="xs:dateTime"
          minOccurs="0"
        />
        <xs:element name="References"
          type="xs:string"
          minOccurs="0"
        />
        <xs:element name="Sender"
          type="t:SingleRecipientType"
          minOccurs="0"
        />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</complexType>
```

```
</xs:complexType>
```

The following table lists the child elements of the PostItemType complex type.

Element name	Type	Description
ConversationIndex	xs:base64Binary ([XMLSCHEMA2])	Contains a binary ID that represents the thread to which this Post object belongs.
ConversationTopic	xs:string ([XMLSCHEMA2])	Represents the conversation identifier.
From	t:SingleRecipientType ([MS-OXWSCDATA] section 2.2.4.69)	Represents the address from which the Post object is sent, and can only be set at creation time.
InternetMessageId	xs:string	Represents the Internet message identifier, as specified in [RFC2822], of the Post object.
IsRead	xs:boolean ([XMLSCHEMA2])	Indicates whether the Post object has been read.
PostedTime	xs:dateTime ([XMLSCHEMA2])	Represents the time at which a Post object is posted. This element is read-only.
References	xs:string	Represents the USENET header , as specified in [RFC850], that is used to associate replies with the Post object.
Sender	t:SingleRecipientType	Identifies the sender of the Post object, and can only be set at creation time.

2.2.4.2 t:PostReplyItemBaseType Complex Type

The **PostReplyItemBaseType** complex type is the base type for the **PostReplyItemType** type, as specified in section 2.2.4.3. The **PostReplyItemBaseType** complex type extends the **ResponseObjectType** complex type, as specified in [MS-OXWSCDATA] section 2.2.4.67.

```
<xs:complexType name="t:PostReplyItemBaseType">
  <xs:complexContent>
    <xs:restriction
      base="t:ResponseObjectType"
    >
      <xs:sequence>
        <xs:element name="Subject"
          type="xs:string"
          minOccurs="0"
        />
        <xs:element name="Body"
          type="t:BodyType"
          minOccurs="0"
        />
        <xs:element name="ReferenceItemId"
          type="t:ItemIdType"
          minOccurs="0"
        />
      </xs:sequence>
      <xs:attribute name="ObjectName"
        type="xs:string"
        use="prohibited"
      />
    </xs:restriction>
  </complexContent>
</complexType>
```

```

    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

```

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

Element name	Type	Description
Subject	xs:string ([XMLSCHEMA2])	Represents the subject of the Post object. The value of the Subject element is limited to 255 characters. Values larger than 255 characters are truncated to the first 252 characters. Three '.' characters are appended to the resulting truncated value.
Body	t:BodyType ([MS-OXWSCDATA] section 2.2.4.19)	Represents the body content of the Post object.
ReferenceItemId	t:ItemIdType ([MS-OXWSCORE] section 2.2.4.24)	Identifies the Post object to which the response refers. This element MUST be present.

The following table lists the attribute of the **PostReplyItemBaseType** complex type.

Attribute name	Type	Description
ObjectName	xs:string ([XMLSCHEMA2])	Represents the name of an object. This name depends on the actual resource accessed.

2.2.4.3 t:PostReplyItemType Complex Type

The **PostReplyItemType** complex type contains a reply to a Post object. The **PostReplyItemType** complex type extends the **PostReplyItemBaseType** complex type, as specified in section [2.2.4.2](#).

```

<xs:complexType>
  <xs:complexContent>
    <xs:extension
      base="t:PostReplyItemBaseType"
    >
      <xs:sequence>
        <xs:element name="NewBodyContent"
          type="t:BodyType"
          minOccurs="0"
        />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

The following table lists the child element of the **PostReplyItemType** complex type.

Element name	Type	Description
NewBodyContent	t:BodyType ([MS-OXWSCDATA] section 2.2.4.19)	Represents the new body content of a Post object.

2.2.5 Simple Types

This specification does not define any common XML schema simple type definitions.

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.

Preliminary

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

This protocol defines a single port type and uses six operations that are specified in [\[MS-OXWSCORE\]](#). These operations enable client implementations to create, retrieve, update, move, copy, and delete Post objects on the server.

3.1.1 Abstract Data Model

None.

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 operations as defined by this specification.

Operation	Description
CopyItem	Copies Post objects on the server.
CreateItem	Creates Post objects on the server.
DeleteItem	Deletes Post objects on the server.
GetItem	Retrieves Post objects from the server.
MoveItem	Moves Post objects on the server.
UpdateItem	Updates Post objects on the server.

3.1.4.1 CopyItem Operation

The **CopyItem** operation copies Post objects on the server. This operation is specified in [\[MS-OXWSCORE\]](#) section 3.1.4.1.

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

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



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

The following is the WSDL binding specification for 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 following table summarizes the set of WSDL message definitions that are specific to the **CopyItem** operation.

Message name	Description
CopyItemSoapIn	<p>The CopyItemSoapIn message is specified in [MS-OXWSCORE] section 3.1.4.1.1.1. It specifies the CopyItem operation request to copy one or more Post objects on the server. The CopyItem element ([MS-OXWSCORE] section 3.1.4.1.2.1) that specifies the XML request MUST contain the following child elements:</p> <p>ToFolderId ([MS-OXWSCORE] section 2.2.4.15)</p> <p>ItemIds ([MS-OXWSCORE] section 2.2.4.15)</p> <p>The ItemIds element MUST contain one or more ItemId child elements ([MS-OXWSCORE] section 2.2.4.30), each one having an Id attribute ([MS-OXWSCORE] section 2.2.4.24) that identifies a Post object.</p> <p>The ItemIds element MUST NOT contain an OccurrenceId child element ([MS-OXWSCORE] section 2.2.4.30) or a RecurringMasterItemId child element ([MS-OXWSCORE] section 2.2.4.30) in a request to copy a Post object.</p>
CopyItemSoapOut	<p>The CopyItemSoapOut message is specified in [MS-OXWSCORE] section 3.1.4.1.1.2. It specifies the server response to the CopyItem operation request.</p> <p>In the response message, the Items element ([MS-OXWSCDATA] section 2.2.4.41) contains one or more PostItem elements of type PostItemType (section 2.2.4.1).</p>

The client sends a **tns:CopyItemSoapIn** request WSDL message and the server MUST respond with a **tns:CopyItemSoapOut** response WSDL message.

3.1.4.2 CreateItem Operation

The **CreateItem** operation creates Post objects on the server. This operation is specified in [MS-OXWSCORE] section 3.1.4.2.

The following is the WSDL port type specification for 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 for 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 following table summarizes the set of WSDL message definitions that are specific to the **CreateItem** operation.

Message name	Description
CreateItemSoapIn	<p>The CreateItemSoapIn message is specified in [MS-OXWSCORE] section 3.1.4.1.1.1. It specifies the CreateItem operation request to create one or more Post objects on the server.</p> <p>The Items child element ([MS-OXWSCDATA] section 2.2.4.46) of the CreateItem element ([MS-OXWSCORE] section 3.1.4.2.2.1) that specifies the XML request MUST contain one or more PostItem elements ([MS-OXWSCDATA] section 2.2.4.46) or PostReplyItem elements ([MS-OXWSCDATA] section 2.2.4.46).</p> <p>The post item MUST not be created in the following folders: Calendar folder, Contacts folder, Tasks folder, Notes folder, Journal folder, Search folder, and Outlook Search folder. Otherwise, the server MUST return an ErrorCannotCreatePostItemInNonMailFolder response code, as specified in [MS-OXWSCDATA] section 2.2.5.23.</p>
1. CreateItemSoapOut	<p>The CreateItemSoapOut message is specified in [MS-OXWSCORE] section 3.1.4.2.1.2. It specifies the server response to the CreateItem operation request.</p> <p>In the response message, the Items element ([MS-OXWSCDATA] section 2.2.4.41) contains one or more PostItem elements ([MS-OXWSCDATA] section 2.2.4.10).</p>

The client sends a **tns:CreateItemSoapIn** request WSDL message and the server MUST respond with a **tns:CreateItemSoapOut** response WSDL message.

3.1.4.3 DeleteItem Operation

The **DeleteItem** operation deletes Post objects on the server. This operation is specified in [\[MS-OXWSCORE\]](#) section 3.1.4.3.

The following is the WSDL port type specification for 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 for 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 following table summarizes the set of WSDL message definitions that are specific to the **DeleteItem** operation.

Message name	Description
DeleteItemSoapIn	<p>The DeleteItemSoapIn message is specified in [MS-OXWSCORE] section 3.1.4.3.1.1. It specifies the DeleteItem operation request to delete one or more Post objects on the server.</p> <p>The DeleteItem element ([MS-OXWSCORE] section 3.1.4.3.2.1) that specifies the XML request MUST contain the ItemIds child element ([MS-OXWSCORE] section 3.1.4.3.3.2).</p> <p>The ItemIds element MUST contain one or more ItemId child elements ([MS-OXWSCORE] section 2.2.4.30), each one having an Id attribute ([MS-OXWSCORE] section 2.2.4.24) that identifies a Post object.</p> <p>The ItemIds element MUST NOT contain an OccurrenceId child element ([MS-OXWSCORE] section 2.2.4.30) or a RecurringMasterItemId child element ([MS-OXWSCORE] section 2.2.4.30) in a request to delete a Post object.</p>
DeleteItemSoapOut	<p>The DeleteItemSoapOut message is specified in [MS-OXWSCORE] section 3.1.4.3.1.2. It specifies the server response to the DeleteItem operation request.</p>

The client sends a **tns>DeleteItemSoapIn** request WSDL message and the server MUST respond with a **tns>DeleteItemSoapOut** response WSDL message.

3.1.4.4 GetItem Operation

The **GetItem** operation retrieves Post objects from the server. This operation is specified in [\[MS-OXWSCORE\]](#) section 3.1.4.4.

The following is the WSDL port type specification for 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 for 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:header message="tns:GetItemSoapIn" part="DateTimePrecision" use="literal"/>
    <soap:header message="tns:GetItemSoapIn" part="ManagementRole" 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 following table summarizes the set of WSDL message definitions that are specific to the **GetItem** operation.

Message name	Description
GetItemSoapIn	<p>The GetItemSoapIn message is specified in [MS-OXWScore] section 3.1.4.4.1.1. It specifies the GetItem operation request to retrieve one or more Post objects from the server.</p> <p>The GetItem element ([MS-OXWScore] section 3.1.4.4.2.1) that specifies the XML request MUST contain the following child elements:</p> <ul style="list-style-type: none"> ItemShape ([MS-OXWScore] section 3.1.4.4.3.2) ItemIds ([MS-OXWScore] section 3.1.4.4.3.2) <p>The ItemIds element MUST contain one or more ItemId child elements ([MS-OXWScore] section 2.2.4.30), each one having an Id attribute ([MS-OXWScore] section 2.2.4.24) that identifies a Post object.</p> <p>The ItemIds element MUST NOT contain an OccurrenceId child element ([MS-OXWScore] section 2.2.4.30) or a RecurringMasterItemId child element ([MS-OXWScore] section 2.2.4.30) in a request to retrieve a Post object.</p>
GetItemSoapOut	<p>The GetItemSoapOut message is specified in [MS-OXWScore] section 3.1.4.4.1.2. It specifies the server response to the GetItem operation request.</p> <p>In the response message, the Items element ([MS-OXWScore] section 2.2.4.41) contains one or more PostItem elements of type PostItemType (section 2.2.4.1).</p>

The client sends a **tns:GetItemSoapIn** request WSDL message and the server MUST respond with a **tns:GetItemSoapOut** response WSDL message.

3.1.4.5 MoveItem Operation

The **MoveItem** operation moves Post objects on the server. This operation is specified in [MS-OXWScore] section 3.1.4.7.

The following is the WSDL port type specification for 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 for 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 following table summarizes the set of WSDL message definitions that are specific to the **MoveItem** operation.

Message name	Description
MoveItemSoapIn	<p>The MoveItemSoapIn message is specified in [MS-OXWSCORE] section 3.1.4.7.1.1. It specifies the MoveItem operation request to move one or more Post objects on the server.</p> <p>The MoveItem element ([MS-OXWSCORE] section 3.1.4.7.2.1) that specifies the XML request MUST contain the following child elements:</p> <p>ToFolderId ([MS-OXWSCORE] section 2.2.4.15)</p> <p>ItemIds ([MS-OXWSCORE] section 2.2.4.15)</p> <p>The ItemIds element MUST contain one or more ItemId child elements ([MS-OXWSCORE] section 2.2.4.30), each one having an Id attribute ([MS-OXWSCORE] section 2.2.4.24) that identifies a Post object.</p> <p>The ItemIds element MUST NOT contain an OccurrenceId child element ([MS-OXWSCORE] section 2.2.4.30) or a RecurringMasterItemId child element ([MS-OXWSCORE] section 2.2.4.30) in a request to move a Post object.</p>
MoveItemSoapOut	<p>The MoveItemSoapOut message is specified in [MS-OXWSCORE] section 3.1.4.7.1.2. It specifies the server response to the MoveItem operation request.</p> <p>In the response message, the Items element ([MS-OXWSCDATA] section 2.2.4.41) contains one or more PostItem elements of type PostItemType (section 2.2.4.1).</p>

The client sends a **tns:MoveItemSoapIn** request WSDL message and the server MUST respond with a **tns:MoveItemSoapOut** response WSDL message.

3.1.4.6 UpdateItem Operation

The **UpdateItem** operation updates Post objects on the server. This operation is specified in [MS-OXWSCORE] section 3.1.4.9.

The following is the WSDL port type specification for the **UpdateItem** operation.

```

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

```

The following is the WSDL binding specification for 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 following table summarizes the set of WSDL message definitions that are specific to the **UpdateItem** operation.

Message name	Description
UpdateItemSoapIn	<p>The UpdateItemSoapIn message is specified in [MS-OXWSCORE] section 3.1.4.9.1.1. It specifies the UpdateItem operation request to update one or more Post objects on the server.</p> <p>The UpdateItem element ([MS-OXWSCORE] section 3.1.4.9.2.1) that specifies the XML request MUST contain an ItemChanges element ([MS-OXWSCORE] section 3.1.4.9.3.2) that contains one or more ItemChange elements ([MS-OXWSCORE] section 3.1.4.9.3.9).</p> <p>Each ItemChange element MUST contain an ItemId child element ([MS-OXWSCORE] section 3.1.4.9.3.7) with an Id attribute ([MS-OXWSCORE] section 2.2.4.24) that identifies the Post object.</p> <p>An ItemChange element MUST NOT contain an OccurrenceId child element ([MS-OXWSCORE] section 3.1.4.9.3.7) or a RecurringMasterItemId child element ([MS-OXWSCORE] section 3.1.4.9.3.7) for a change to a Post object.</p> <p>Each AppendToItemField element ([MS-OXWSCORE] section 3.1.4.9.3.8) and SetItemField element ([MS-OXWSCORE] section 3.1.4.9.3.8) in the request MUST contain a PostItem element of type PostItemType (section 2.2.4.1).</p>
UpdateItemSoapOut	<p>The UpdateItemSoapOut message is specified in [MS-OXWSCORE] section 3.1.4.9.1.2. It specifies the server response to the UpdateItem operation request.</p> <p>In the response message, the Items element ([MS-OXWSCDATA] section 2.2.4.41) contains one or more PostItem elements of type PostItemType.</p>

The client sends a **tns:UpdateItemSoapIn** request WSDL message and the server MUST respond with a **tns:UpdateItemSoapOut** response WSDL message.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

4 Protocol Examples

In the following examples, Jason Carlson is using a protocol client to manage Post objects in a **public folder** named "Announcements". Jason's company holds a company-wide meeting every three months, and Jason wants to announce the dates for the meetings in this public folder.

4.1 Copying a Post Object

In this scenario, Jason wants to create a copy of the Post object that he created for the July 22 meeting so that he can later modify it to announce the next company meeting on October 21.

The following is the client request to copy the Post object.

```
<?xml version="1.0" encoding="utf-16"?>
<CopyItemType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ToFolderId xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <FolderId Id="AQEuAAADGkRzkKpmEc2byACqAC/EWgMatzciqtvmvE0GtSfqReMVLgABEfqpawAAAA=="
ChangeKey="AQAAAABYAAAC3NyKq2+a8Q4a1J+pF4xUuAAER+pqp"
xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
  </ToFolderId>
  <ItemIds xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <ItemId
Id="AQIARgAAAxpEc5CqZhHNm8gAqgAvxFoJALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAC3NyKq2+a8Q4a1J+pF4xUuA
AER+pqyAAAAALgAAAxpEc5CqZhHNm8gAqgAvxFoDALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAA="
ChangeKey="FgAAAABYAAAC3NyKq2+a8Q4a1J+pF4xUuAAER+py4"
xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
    </ItemId>
  </ItemIds>
  <ReturnNewItemIds
xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">true</ReturnNewItemIds>
</CopyItemType>
```

The following is the server response.

```
<?xml version="1.0" encoding="utf-16"?>
<CopyItemResponseType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ResponseMessages xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <CopyItemResponseMessage ResponseClass="Success">
      <ResponseCode>NoError</ResponseCode>
      <Items>
        <PostItem xmlns="http://schemas.microsoft.com/exchange/services/2006/types">
          <ItemId
Id="AQIARgAAAxpEc5CqZhHNm8gAqgAvxFoJALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAC3NyKq2+a8Q4a1J+pF4xUuA
AER+pqzAAAAALgAAAxpEc5CqZhHNm8gAqgAvxFoDALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAA="
ChangeKey="FgAAAA==" />
          </PostItem>
        </Items>
      </CopyItemResponseMessage>
    </ResponseMessages>
  </CopyItemResponseType>
```

4.2 Creating a Post Object

In this scenario, Jason wants to create a new Post object to announce the upcoming company meeting on July 22.

The following is the client request to create a new Post object.

```

<?xml version="1.0" encoding="utf-16"?>
<CreateItemType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" MessageDisposition="SaveOnly">
  <SavedItemFolderId xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <FolderId Id="AQEuAAADGkRzkKpmEc2byACqAC/EWgMatzciqtvmvEOGtSfqReMVLgABEFqapwAAAA=="
ChangeKey="AQAAABYAAAC3NyKq2+a8Q4a1J+pF4xUuAAER+pqp"
xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
  </SavedItemFolderId>
  <Items xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <PostItem xmlns="http://schemas.microsoft.com/exchange/services/2006/types">
      <Subject>Company meeting scheduled for July 22</Subject>
      <Body BodyType="HTML">Please see www.contoso.com/companymeeting for full
details.</Body>
      <From>
        <Mailbox>
          <EmailAddress>jason@contoso.com</EmailAddress>
        </Mailbox>
      </From>
    </PostItem>
  </Items>
</CreateItemType>

```

The following is the server response.



```

<?xml version="1.0" encoding="utf-16"?>
<CreateItemResponseType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ResponseMessages xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <CreateItemResponseMessage ResponseClass="Success">
      <ResponseCode>NoError</ResponseCode>
      <Items>
        <PostItem xmlns="http://schemas.microsoft.com/exchange/services/2006/types">
          <ItemId>
            Id="AQIARgAAAxpEc5CqZhHNm8gAqgAvxFoJALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAC3NyKq2+a8Q4a1J+pF4xUuA
AER+pqyAAAAALgAAAxpEc5CqZhHNm8gAqgAvxFoDALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAA="
ChangeKey="FgAAABYAAAC3NyKq2+a8Q4a1J+pF4xUuAAER+py4" />
          </PostItem>
        </Items>
      </CreateItemResponseMessage>
    </ResponseMessages>
  </CreateItemResponseType>

```

4.3 Deleting a Post Object

In this scenario, Jason wants to delete the Post object from the "Archive" public folder.

The following is the client request to delete the Post object.

```

<?xml version="1.0" encoding="utf-16"?>
<DeleteItemType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" DeleteType="HardDelete">
  <ItemIds xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <ItemId>
      Id="AQIARgAAAxpEc5CqZhHNm8gAqgAvxFoJALc3Iqrb5rxDhrUn6kXjFS4AARH6nq4AAAC3NyKq2+a8Q4a1J+pF4xUuA
AER+p7GAAAAALgAAAxpEc5CqZhHNm8gAqgAvxFoDALc3Iqrb5rxDhrUn6kXjFS4AARH6nq4AAAA="
ChangeKey="FgAAABYAAAC3NyKq2+a8Q4a1J+pF4xUuAAER+qDG"
xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
    </ItemIds>
  </DeleteItemType>

```

The following is the server response.


```

<?xml version="1.0" encoding="utf-16"?>
<DeleteItemResponseType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ResponseMessages xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <DeleteItemResponseMessage ResponseClass="Success">
      <ResponseCode>NoError</ResponseCode>
    </DeleteItemResponseMessage>
  </ResponseMessages>
</DeleteItemResponseType>

```

4.4 Moving a Post Object

In this scenario, Jason wants to move the Post object announcing the July 22 meeting to a public folder named "Archive".

The following is the client request to move the Post object.

```

<?xml version="1.0" encoding="utf-16"?>
<MoveItemType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ToFolderId xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <FolderId Id="AQEuAAADGkRzkKpmEc2byACqAC/EWgMATzciqtvmvEOGtSfqReMVLgABEfqergAAAA=="
ChangeKey="AQAAAABYAAAC3NyKq2+a8Q4a1J+pF4xUuAAER+p6w"
xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
  </ToFolderId>
  <ItemIds xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <ItemId
Id="AQIARgAAAxpEc5CqZhHNm8gAqgAvxFoJALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAC3NyKq2+a8Q4a1J+pF4xUuA
AER+pqyAAAALgAAAxpEc5CqZhHNm8gAqgAvxFoDALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAA="
ChangeKey="FgAAAABYAAAC3NyKq2+a8Q4a1J+pF4xUuAAER+py6"
xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
  </ItemIds>
  <ReturnNewItemIds
xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">true</ReturnNewItemIds>
</MoveItemType>

```

The following is the server response.

```

<?xml version="1.0" encoding="utf-16"?>
<MoveItemResponseType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ResponseMessages xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <MoveItemResponseMessage ResponseClass="Success">
      <ResponseCode>NoError</ResponseCode>
      <Items>
        <PostItem xmlns="http://schemas.microsoft.com/exchange/services/2006/types">
          <ItemId
Id="AQIARgAAAxpEc5CqZhHNm8gAqgAvxFoJALc3Iqrb5rxDhrUn6kXjFS4AARH6nq4AAAC3NyKq2+a8Q4a1J+pF4xUuA
AER+p7GAAAALgAAAxpEc5CqZhHNm8gAqgAvxFoDALc3Iqrb5rxDhrUn6kXjFS4AARH6nq4AAAA="
ChangeKey="FgAAAA==" />
        </PostItem>
      </Items>
    </MoveItemResponseMessage>
  </ResponseMessages>
</MoveItemResponseType>

```

4.5 Retrieving a Post Object

In this scenario, Jason wants to view the new Post object that he just created to make sure that there are no errors.

The following is the client request to retrieve the Post object from the server.

```
<?xml version="1.0" encoding="utf-16"?>
<GetItemType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ItemShape xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <BaseShape
xmlns="http://schemas.microsoft.com/exchange/services/2006/types">Default</BaseShape>
  </ItemShape>
  <ItemIds xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <ItemId
Id="AQIARgAAAXpEc5CqZhhNm8gAqgAvxFoJALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAC3NyKq2+a8Q4a1J+pF4xUuA
AER+pqyAAAAALgAAAXpEc5CqZhhNm8gAqgAvxFoDALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAA="
ChangeKey="FgAAABYAAAC3NyKq2+a8Q4a1J+pF4xUuAAER+py4"
xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
  </ItemIds>
</GetItemType>
```

The following is the server response.

```
<?xml version="1.0" encoding="utf-16"?>
<GetItemResponseType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ResponseMessages xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <GetItemResponseMessage ResponseClass="Success">
      <ResponseCode>NoError</ResponseCode>
      <Items>
        <PostItem xmlns="http://schemas.microsoft.com/exchange/services/2006/types">
          <ItemId
Id="AQIARgAAAXpEc5CqZhhNm8gAqgAvxFoJALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAC3NyKq2+a8Q4a1J+pF4xUuA
AER+pqyAAAAALgAAAXpEc5CqZhhNm8gAqgAvxFoDALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAA="
ChangeKey="FgAAABYAAAC3NyKq2+a8Q4a1J+pF4xUuAAER+py4" />
          <Subject>Company meeting scheduled for July 22</Subject>
          <HasAttachments>>false</HasAttachments>
          <ConversationIndex>AcxcN9djfXVXymjrQSGXSyVfDhtJJA==</ConversationIndex>
          <ConversationTopic>Company meeting scheduled for July 22</ConversationTopic>
          <From>
            <Mailbox>
              <Name>Jason Carlson</Name>
              <EmailAddress>jason@contoso.com</EmailAddress>
              <RoutingType>SMTP</RoutingType>
              <MailboxType>Mailbox</MailboxType>
            </Mailbox>
          </From>
        </PostItem>
      </Items>
    </GetItemResponseMessage>
  </ResponseMessages>
</GetItemResponseType>
```

4.6 Updating a Post Object

In this scenario, Jason wants to update the copy of the original Post object that was created in the example in section [4.1](#) to announce the next company meeting on October 21.

The following is the client request to update the Post object.

```
<?xml version="1.0" encoding="utf-16"?>
<UpdateItemType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" ConflictResolution="AlwaysOverwrite">
  <ItemChanges xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <ItemChange xmlns="http://schemas.microsoft.com/exchange/services/2006/types">
      <ItemId
Id="AQIARgAAAxpEc5CqZhhNm8gAqgAvxFoJALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAC3NyKq2+a8Q4a1J+pF4xUuA
AER+pqyAAAALgAAAxpEc5CqZhhNm8gAqgAvxFoDALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAA="
ChangeKey="FgAAABYAAAC3NyKq2+a8Q4a1J+pF4xUuAAER+py4" />
      <Updates>
        <SetItemField>
          <FieldURI FieldURI="item:Subject" />
          <PostItem>
            <Subject>Company meeting scheduled for October 21</Subject>
          </PostItem>
        </SetItemField>
      </Updates>
    </ItemChange>
  </ItemChanges>
</UpdateItemType>
```

The following is the server response.

```
<?xml version="1.0" encoding="utf-16"?>
<UpdateItemResponseType xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ResponseMessages xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <UpdateItemResponseMessage ResponseClass="Success">
      <ResponseCode>NoError</ResponseCode>
      <Items>
        <PostItem xmlns="http://schemas.microsoft.com/exchange/services/2006/types">
          <ItemId
Id="AQIARgAAAxpEc5CqZhhNm8gAqgAvxFoJALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAC3NyKq2+a8Q4a1J+pF4xUuA
AER+pqyAAAALgAAAxpEc5CqZhhNm8gAqgAvxFoDALc3Iqrb5rxDhrUn6kXjFS4AARH6mqcAAAA="
ChangeKey="FgAAABYAAAC3NyKq2+a8Q4a1J+pF4xUuAAER+py6" />
        </PostItem>
      </Items>
      <ConflictResults>
        <Count xmlns="http://schemas.microsoft.com/exchange/services/2006/types">0</Count>
      </ConflictResults>
    </UpdateItemResponseMessage>
  </ResponseMessages>
</UpdateItemResponseType>
```

5 Security

5.1 Security Considerations for Implementers

None.

5.2 Index of Security Parameters

None.

Preliminary

6 Appendix A: Full WSDL

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

File name	Description	Section
MS-OXWSPOST.wsdl	Contains the WSDL for the implementation of this protocol.	6
MS-OXWSCORE-messages.xsd	Contains the XML schema message definitions that are used in this protocol.	[MS-OXWSCORE] section 7.1
MS-OXWSPOST-types.xsd	Contains the XML schema type definitions that are used in this protocol.	7.2

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-OXWSPOST-types.xsd or MS-OXWSCORE-messages.xsd schemas have to be placed in the common folder with these files.

This section contains the contents of the MS-OXWSPOST.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="Exchange2016"
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="Exchange2016"
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="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="CreateItemSoapIn">
    <wsdl:part name="request" element="tns:CreateItem"/>
  </wsdl:message>
</wsdl:definitions>
```

```

    <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="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: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="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:portType name="ExchangeServicePortType">
  <wsdl:operation name="GetItem">
    <wsdl:input message="tns:GetItemSoapIn"/>
    <wsdl:output message="tns:GetItemSoapOut"/>
  </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="UpdateItem">
    <wsdl:input message="tns:UpdateItemSoapIn"/>
    <wsdl:output message="tns:UpdateItemSoapOut"/>
  </wsdl:operation>
  <wsdl:operation name="MoveItem">

```

```

        <wsdl:input message="tns:MoveItemSoapIn"/>
        <wsdl:output message="tns:MoveItemSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="CopyItem">
        <wsdl:input message="tns:CopyItemSoapIn"/>
        <wsdl:output message="tns:CopyItemSoapOut"/>
    </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="ExchangeServiceBinding" type="tns:ExchangeServicePortType">
    <wsdl:documentation>
        <wsi:Claim conformsTo="http://ws-i.org/profiles/basic/1.0" xmlns:wsi="http://ws-
i.org/schemas/conformanceClaim"/>
    </wsdl:documentation>
    <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
    <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:header message="tns:GetItemSoapIn" part="DateTimePrecision" use="literal"/>
            <soap:header message="tns:GetItemSoapIn" part="ManagementRole" 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="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="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="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="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:binding>
</wsdl:definitions>

```

Prell

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
Messages schema	m:	[MS-OXWSCORE] section 7.1
Types schema	t:	7.2

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-OXWSPPOST-types.xsd or MS-OXWSCORE-messages.xsd schemas have to be placed in the common folder along with the files listed in the table.

7.1 Messages Schema

This protocol uses the XML schema message definitions in MS-OXWSCORE-messages.xsd, as described in [\[MS-OXWSCORE\]](#) section 7.1.

7.2 Types Schema

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

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

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

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema 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/types"
elementFormDefault="qualified" version="Exchange2016" id="types">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xs:include schemaLocation="MS-OXWSCORE-types.xsd"/>

  <xs:complexType name="PostItemType">
    <xs:complexContent>
      <xs:extension base="t:ItemType">
        <xs:sequence>
          <xs:element name="ConversationIndex" type="xs:base64Binary" minOccurs="0"/>
          <xs:element name="ConversationTopic" type="xs:string" minOccurs="0"/>
          <!-- From property can only be set at creation time -->
          <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" />
          <!-- PostedTime is read only -->
          <xs:element name="PostedTime" type="xs:dateTime" minOccurs="0"/>
          <xs:element name="References" type="xs:string" minOccurs="0" />
          <!-- Sender can only be set at creation time -->
          <xs:element name="Sender" type="t:SingleRecipientType" minOccurs="0"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

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

  <xs:complexType name="PostReplyItemBaseType">
    <xs:complexContent>
      <xs:restriction base="t:ResponseObjectType">
        <xs:sequence>
          <xs:element name="Subject" type="xs:string" minOccurs="0" />
          <xs:element name="Body" type="t:BodyType" minOccurs="0" />
          <xs:element name="ReferenceItemId" type="t:ItemIdType" minOccurs="0" />
        </xs:sequence>
        <xs:attribute name="ObjectName" type="xs:string" use="prohibited" />
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="PostReplyItemType">
    <xs:complexContent>
      <xs:extension base="t:PostReplyItemBaseType">
        <xs:sequence>
          <xs:element name="NewBodyContent" type="t:BodyType" minOccurs="0" />
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>
```

Preliminary

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 Service Pack 1 (SP1)
- Microsoft Exchange Server 2010
- Microsoft Exchange Server 2013
- Microsoft Exchange Server 2016 Preview

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 this document since the last release. 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.
- The removal of a document from the documentation set.

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 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 changes were introduced. Minor editorial and formatting changes may have been made, but the technical content of the document is identical to the last released version.

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.
- 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 dochelp@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
8 Appendix C: Product Behavior	Added Exchange 2016 to list of applicable products.	Y	Content update.

Preliminary

10 Index

A

Abstract data model
[server](#) 16
[Applicability](#) 9
[Attribute groups](#) 15
[Attributes](#) 15

C

[Capability negotiation](#) 9
[Change tracking](#) 36
[Complex types](#) 12
 [t:PostItemType Complex Type](#) 12
 [t:PostReplyItemBaseType Complex Type](#) 13
 [t:PostReplyItemType Complex Type](#) 14
[Copying a Post object example](#) 23
[Creating a Post object example](#) 23

D

Data model - abstract
[server](#) 16
[Deleting a Post object example](#) 24

E

Events
 [local - server](#) 22
 [timer - server](#) 22
Examples
 [copying a Post object](#) 23
 [creating a Post object](#) 23
 [deleting a Post object](#) 24
 [moving a Post object](#) 25
 [overview](#) 23
 [retrieving a Post object](#) 25
 [updating a Post object](#) 27

F

[Fields - vendor-extensible](#) 9
[Full WSDL](#) 29
[Full XML schema](#) 33
 [Messages Schema](#) 33
 [Types Schema](#) 33

G

[Glossary](#) 6
[Groups](#) 15

I

[Implementer - security considerations](#) 28
[Index of security parameters](#) 28
[Informative references](#) 8
Initialization
 [server](#) 16

[Introduction](#) 6

L

Local events
 [server](#) 22

M

Message processing
 [server](#) 16
Messages
 [attribute groups](#) 15
 [attributes](#) 15
 [complex types](#) 12
 [elements](#) 11
 [enumerated](#) 11
 [groups](#) 15
 [namespaces](#) 11
 [simple types](#) 15
 [syntax](#) 11
 [t:PostItemType Complex Type complex type](#) 12
 [t:PostReplyItemBaseType Complex Type complex type](#) 13
 [t:PostReplyItemType Complex Type complex type](#) 14
 [transport](#) 11
[Moving a Post object example](#) 25

N

[Namespaces](#) 11
[Normative references](#) 7

O

Operations
 [CopyItem Operation](#) 16
 [CreateItem Operation](#) 17
 [DeleteItem Operation](#) 18
 [GetItem Operation](#) 19
 [MoveItem Operation](#) 20
 [UpdateItem Operation](#) 21
[Overview \(synopsis\)](#) 8

P

[Parameters - security index](#) 28
[Preconditions](#) 9
[Prerequisites](#) 9
[Product behavior](#) 35
Protocol Details
 [overview](#) 16

R

[References](#) 7
 [informative](#) 8
 [normative](#) 7

[Relationship to other protocols](#) 8
[Retrieving a Post object example](#) 25

S

Security
[implementer considerations](#) 28
[parameter index](#) 28

Sequencing rules
[server](#) 16

Server
[abstract data model](#) 16
[CopyItem Operation operation](#) 16
[CreateItem Operation operation](#) 17
[DeleteItem Operation operation](#) 18
[GetItem Operation operation](#) 19
[initialization](#) 16
[local events](#) 22
[message processing](#) 16
[MoveItem Operation operation](#) 20
[sequencing rules](#) 16
[timer events](#) 22
[timers](#) 16
[UpdateItem Operation operation](#) 21

[Simple types](#) 15

[Standards assignments](#) 10

Syntax
[messages - overview](#) 11

T

[t:PostItemType Complex Type complex type](#) 12
[t:PostReplyItemBaseType Complex Type complex type](#) 13
[t:PostReplyItemType Complex Type complex type](#) 14

Timer events
[server](#) 22

Timers
[server](#) 16

[Tracking changes](#) 36

[Transport](#) 11

Types
[complex](#) 12
[simple](#) 15

U

[Updating a Post object example](#) 27

V

[Vendor-extensible fields](#) 9
[Versioning](#) 9

W

[WSDL](#) 29

X

[XML schema](#) 33
[Messages Schema](#) 33
[Types Schema](#) 33