

[MS-OXWSARCH]: Archiving Web Service Protocol Specification

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft [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.
- **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 (beta) 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 (beta) 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.

Revision Summary

Date	Revision History	Revision Class	Comments
07/16/2012	0.1	New	Released new document.

Table of Contents

1 Introduction	5
1.1 Glossary	5
1.2 References	5
1.2.1 Normative References	5
1.2.2 Informative References	6
1.3 Overview	6
1.4 Relationship to Other Protocols	6
1.5 Prerequisites/Preconditions	7
1.6 Applicability Statement	7
1.7 Versioning and Capability Negotiation	7
1.8 Vendor-Extensible Fields	7
1.9 Standards Assignments	8
2 Messages	9
2.1 Transport	9
2.2 Common Message Syntax	9
2.2.1 Namespaces	9
2.2.2 Messages	9
2.2.3 Elements	9
2.2.4 Complex Types	9
2.2.5 Simple Types	10
2.2.6 Attributes	10
2.2.7 Groups	10
2.2.8 Attribute Groups	10
3 Protocol Details	11
3.1 ExchangeServicePortType Server Details	11
3.1.1 Abstract Data Model	11
3.1.2 Timers	11
3.1.3 Initialization	11
3.1.4 Message Processing Events and Sequencing Rules	11
3.1.4.1 ArchiveItem	11
3.1.4.1.1 Messages	12
3.1.4.1.1.1 ArchiveItemSoapIn	12
3.1.4.1.1.2 ArchiveItemSoapOut	13
3.1.4.1.2 Elements	13
3.1.4.1.2.1 ArchiveItem	13
3.1.4.1.2.2 ArchiveItemResponse	14
3.1.4.1.3 Complex Types	14
3.1.4.1.3.1 ArchiveItemType	14
3.1.4.1.3.2 ArchiveItemResponseType	14
3.1.4.1.4 Simple Types	15
3.1.4.1.5 Attributes	15
3.1.4.1.6 Groups	15
3.1.4.1.7 Attribute Groups	15
3.1.4.2 CreateFolderPath	15
3.1.4.2.1 Messages	16
3.1.4.2.1.1 CreateFolderPathSoapIn	16
3.1.4.2.1.2 CreateFolderPathSoapOut	17
3.1.4.2.2 Elements	17

3.1.4.2.2.1	CreateFolderPath	17
3.1.4.2.2.2	CreateFolderPathResponse	17
3.1.4.2.3	Complex Types	18
3.1.4.2.3.1	CreateFolderPathType	18
3.1.4.2.3.2	CreateFolderPathResponseType	18
3.1.4.2.4	Simple Types	18
3.1.4.2.5	Attributes	18
3.1.4.2.6	Groups	19
3.1.4.2.7	Attribute Groups	19
3.1.5	Timer Events	19
3.1.6	Other Local Events	19
4	Protocol Examples	20
5	Security	21
5.1	Security Considerations for Implementers	21
5.2	Index of Security Parameters	21
6	Appendix A: Full WSDL	22
7	Appendix B: Full XML Schema	24
8	Appendix C: Product Behavior	25
9	Change Tracking	26
10	Index	27

1 Introduction

The Archiving Web Service Protocol enables clients to use a **Web service** to implement item-level archiving of 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)

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

endpoint
mailbox
Simple Object Access Protocol (SOAP)
SOAP action
SOAP body
SOAP header
Web service
Web Services Description Language (WSDL)
WSDL message
WSDL operation
WSDL port type
XML namespace
XML namespace prefix
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-OXWSCDATA] Microsoft Corporation, "[Common Web Service Data Types](#)".

[MS-OXWSGTZ] Microsoft Corporation, "[Get Server Time Zone 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>

[RFC3066] Alvestrand, H., "Tags for the Identification of Language", RFC 3066, January 2001, <http://www.ietf.org/rfc/rfc3066.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-OXDCLI] Microsoft Corporation, "[Autodiscover Publishing and Lookup Protocol Specification](#)".

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

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

[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

Archive mailboxes are secondary mailboxes that are associated with a user. Archive mailboxes are typically used to manage email storage limits. For example, older email items might periodically be moved from the Inbox to the archive mailbox.

This protocol provides operations to archive a set of mail items from a primary mailbox. Archiving Inbox items in this way preserves the folder hierarchy of the items. In addition, archive mailboxes can now be stored either locally on a client, or remotely, in a way that is mostly opaque to a user, by using a folder path to point to the contents of the archive.

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 [\[MS-OXDCLI\]](#), 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 that is exchanged between the client and the server. This protocol uses the XML Protocol, as described in [\[XMLSCHEMA1\]](#) and [\[XMLSCHEMA2\]](#), to describe the message content that is 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 layering diagram.

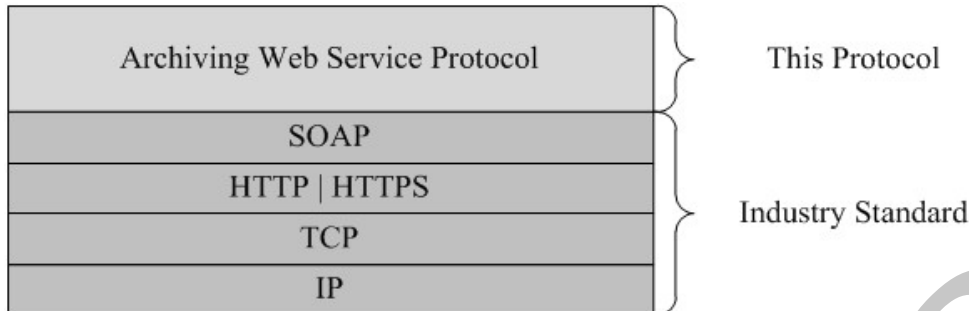


Figure 1: This protocol in relation to other protocols

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 use Web services to implement archiving from a user's mailbox. This protocol is applicable to all SOAP-based clients, as described in [\[SOAP1.1\]](#).

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

- **Supported Transports:** This protocol uses SOAP 1.1, as described in section [2.1](#).
- **Protocol Versions:** This protocol describes 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.
- **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

The SOAP version used for this protocol is SOAP 1.1, as specified in [\[SOAP1.1\]](#).

This protocol relies on the web server that hosts the application to perform authentication. The protocol SHOULD use secure communications by means of 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 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** 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 namespace prefix is implementation-specific and not significant for interoperability.

Prefix	Namespace URI	Reference
m	http://schemas.microsoft.com/exchange/services/2006/messages	
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
t	http://schemas.microsoft.com/exchange/services/2006/types	
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
wsi	http://ws-i.org/schemas/conformanceClaim/	
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1] [XMLSCHEMA2]

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

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

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

In the following sections, the schema definition might be less restrictive than the processing rules imposed by the protocol. The WSDL in this specification matches the WSDL that shipped with the product and provides a base description of the schema. The text that introduces the WSDL specifies additional restrictions that reflect actual Microsoft product 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 and present.

3.1 ExchangeServicePortType Server Details

This protocol defines a single WSDL port type.

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
ArchiveItem	The ArchiveItem operation moves an item from the primary mailbox to the archive mailbox.
CreateFolderPath	The CreateFolderPath operation creates a URI that points to the storage location for the archive mailbox.

3.1.4.1 ArchiveItem

The **ArchiveItem** operation moves an item from the primary mailbox to the archive mailbox.

```
<wsdl:operation name="ArchiveItem" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="m:ArchiveItemSoapIn"/>
  <wsdl:output message="m:ArchiveItemSoapOut"/>
</wsdl:operation>
```

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

```
<wsdl:operation name="ArchiveItem" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <soap:operation
    soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/ArchiveItem"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
  <wsdl:input>
```

```

    <soap:header message="m:ArchiveItemSoapIn" part="Impersonation" use="literal"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
    <soap:header message="m:ArchiveItemSoapIn" part="MailboxCulture" use="literal"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
    <soap:header message="m:ArchiveItemSoapIn" part="RequestVersion" use="literal"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
    <soap:body use="literal" parts="request"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal" parts="ArchiveItemResult"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
    <soap:header message="m:ArchiveItemSoapOut" part="ServerVersion" use="literal"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
  </wsdl:output>
</wsdl:operation>

```

3.1.4.1.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
ArchiveItemSoapIn	Specifies the request to archive an item.
ArchiveItemSoapOut	Specifies the response to the ArchiveItem request.

3.1.4.1.1.1 ArchiveItemSoapIn

The **ArchiveItemSoapIn** WSDL message specifies the request to archive an item.

The following is the **ArchiveItem** WSDL message specification.

```

<wsdl:message name="ArchiveItemSoapIn" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:part name="request" element="m:ArchiveItem"/>
  <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
  <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
  <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
</wsdl:message>

```

The **ArchiveItem** WSDL message is the input message for the **SOAP action** <http://schemas.microsoft.com/exchange/services/2006/messages/ArchiveItem>.

The parts of the **ArchiveItem** WSDL message are described in the following table.

Part name	Element/type	Description
request	m:ArchiveItem (section 3.1.4.1.2.1)	Specifies the SOAP body of the request containing the information that is required to query for availability.
Impersonation	t:ExchangeImpersonation ([MS-OXWSCDATA] section 2.2.5.3)	Specifies a SOAP header that identifies the user whom the client is impersonating. <1>

Part name	Element/type	Description
MailboxCulture	t:MailboxCulture ([MS-OXWSCDATA] section 2.2.4.39)	Specifies a SOAP header that identifies the culture to be used for accessing the mailbox. The cultures are defined in [RFC3066] .
RequestVersion	t:RequestServerVersion ([MS-OXWSCDATA] section 2.2.5.9)	Specifies a SOAP header that identifies the schema version for the ArchiveItem operation request. <2>

3.1.4.1.1.2 ArchiveItemSoapOut

The **ArchiveItemSoapIn** WSDL message specifies the response to the **ArchiveItem** request.

The following is the **ArchiveItem** WSDL message specification.

```
<wsdl:message name="ArchiveItemSoapOut" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:part name="ArchiveItemResult" element="m:ArchiveItemResponse"/>
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
```

The **ArchiveItem** WSDL message is the output message for the SOAP action <http://schemas.microsoft.com/exchange/services/2006/messages/ArchiveItem>.

The parts of the **ArchiveItem** WSDL message are described in the following table.

Part name	Element/type	Description
ArchiveItemResult	m:ArchiveItemResponse (section 3.1.4.1.2.2)	Specifies the SOAP body of the response that contains the requested availability information.
ServerVersion	t:ServerVersionInfo ([MS-OXWSCDATA] section 2.2.5.10)	Specifies a SOAP header that identifies the server version for the response.

3.1.4.1.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ArchiveItem	The input data for the ArchiveItem WSDL operation .
ArchiveItemResponse	The result data for the ArchiveItem WSDL operation .

3.1.4.1.2.1 ArchiveItem

The input data for the **ArchiveItem** WSDL operation.

```
<xs:element name="ArchiveItem" type="m:ArchiveItemType"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"/>
```

3.1.4.1.2.2 ArchiveItemResponse

The result data for the **ArchiveItem** WSDL operation.

```
<xs:element name="ArchiveItemResponse" type="m:ArchiveItemResponseType"
xmlns:xs="http://www.w3.org/2001/XMLSchema"/>
```

3.1.4.1.3 Complex Types

The following table summarizes the XML schema complex type definitions that are specific to this operation.

Complex type	Description
ArchiveItemType	Specifies a request to move items from the primary mailbox to an archive folder.
ArchiveItemResponseType	Specifies the response to the ArchiveItem operation.

3.1.4.1.3.1 ArchiveItemType

Namespace: http://schemas.microsoft.com/exchange/services/2006/messages

The **ArchiveItemType** complex type specifies a request to move items from the primary mailbox to an archive folder.

```
<xs:complexType name="ArchiveItemType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexContent mixed="false">
    <xs:extension base="m:BaseRequestType">
      <xs:sequence>
        <xs:element name="ArchiveSourceFolderId" type="t:TargetFolderIdType"/>
        <xs:element name="ItemIds" type="t:NonEmptyArrayOfBaseItemIdsType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

ArchiveSourceFolderId: Specifies the identifier of the source folder.

ItemIds: Specifies the item identifiers.

3.1.4.1.3.2 ArchiveItemResponseType

Namespace: http://schemas.microsoft.com/exchange/services/2006/messages

The **ArchiveItemResponseType** complex type specifies the response for an **ArchiveItem** operation.

```
<xs:complexType name="ArchiveItemResponseType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexContent mixed="false">
    <xs:extension base="m:BaseResponseMessageType"/>
  </xs:complexContent>
</xs:complexType>
```

3.1.4.1.4 Simple Types

None.

3.1.4.1.5 Attributes

None.

3.1.4.1.6 Groups

None.

3.1.4.1.7 Attribute Groups

None.

3.1.4.2 CreateFolderPath

The **CreateFolderPath** operation creates a URI that points to the storage location for the archive mailbox.

```
<wsdl:operation name="CreateFolderPath" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="m:CreateFolderPathSoapIn"/>
  <wsdl:output message="m:CreateFolderPathSoapOut"/>
</wsdl:operation>
```

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

```
<wsdl:operation name="CreateFolderPath" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <soap:operation
    soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/CreateFolderPath"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
  <wsdl:input>
    <soap:header message="m:CreateFolderPathSoapIn" part="Impersonation" use="literal"
      xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
    <soap:header message="m:CreateFolderPathSoapIn" part="MailboxCulture" use="literal"
      xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
    <soap:header message="m:CreateFolderPathSoapIn" part="RequestVersion" use="literal"
      xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
    <soap:header message="m:CreateFolderPathSoapIn" part="TimeZoneContext" use="literal"
      xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
    <soap:body use="literal" parts="request"
      xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal" parts="CreateFolderPathResult"
      xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
    <soap:header message="m:CreateFolderPathSoapOut" part="ServerVersion" use="literal"
      xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
  </wsdl:output>
</wsdl:operation>
```

3.1.4.2.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
CreateFolderPathSoapIn	Specifies the request to create a folder path.
CreateFolderPathSoapOut	Specifies the response to the CreateFolderPathSoapIn request.

3.1.4.2.1.1 CreateFolderPathSoapIn

The **CreateFolderPathSoapIn** WSDL message specifies the request to create a folder path.

The following is the **CreateFolderPath** WSDL message specification.

```
<wsdl:message name="CreateFolderPathSoapIn" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:part name="request" element="m:CreateFolderPath"/>
  <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>
```

The **CreateFolderPath** WSDL message is the input message for the SOAP action <http://schemas.microsoft.com/exchange/services/2006/messages/CreateFolderPath>.

The parts of the **CreateFolderPath** WSDL message are described in the following table.

Part name	Element/type	Description
request	m:CreateFolderPath (section 3.1.4.2.2.1)	Specifies the SOAP body of the request containing the information that is required to query for availability.
Impersonation	t:ExchangeImpersonation ([MS-OXWSCDATA] section 2.2.5.3)	Specifies a SOAP header that identifies the user whom the client is impersonating. <3>
MailboxCulture	t:MailboxCulture ([MS-OXWSCDATA] section 2.2.4.39)	Specifies a SOAP header that identifies the culture to be used for accessing the mailbox. The cultures are defined in [RFC3066] .
RequestVersion	t:RequestServerVersion ([MS-OXWSCDATA] section 2.2.5.9)	Specifies a SOAP header that identifies the schema version for the CreateFolderPath operation request. <4>
TimeZoneContext	t:TimeZoneContext ([MS-OXWSGTZ] section 2.2.3.4)	Specifies a SOAP header that identifies the time zone to use for all responses from the server. All times that are returned from the server will be converted to the specified time zone. <5>

3.1.4.2.1.2 CreateFolderPathSoapOut

The **CreateFolderSoapOut** WSDL message specifies the response to the **CreateFolderPathSoapIn** request.

The following is the **CreateFolderPath** WSDL message specification.

```
<wsdl:message name="CreateFolderPathSoapOut" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:part name="CreateFolderPathResult" element="m:CreateFolderPathResponse"/>
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
```

The **CreateFolderPath** WSDL message is the output message for the SOAP action <http://schemas.microsoft.com/exchange/services/2006/messages/CreateFolderPath>.

The parts of the **CreateFolderPath** WSDL message are described in the following table.

Part name	Element/type	Description
CreateFolderPathResult	m:CreateFolderPathResponse (section 3.1.4.2.2.2)	Represents the result of this operation.
ServerVersion	t:ServerVersionInfo ([MS-OXWSCDATA] section 2.2.5.10)	Specifies a SOAP header that identifies the server version for the response.

3.1.4.2.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
CreateFolderPath	The input data for the CreateFolderPath WSDL operation.
CreateFolderPathResponse	The result data for the CreateFolderPath WSDL operation.

3.1.4.2.2.1 CreateFolderPath

The input data for the **CreateFolderPath** WSDL operation.

```
<xs:element name="CreateFolderPath" type="m:CreateFolderPathType"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"/>
```

3.1.4.2.2.2 CreateFolderPathResponse

The result data for the **CreateFolderPath** WSDL operation.

```
<xs:element name="CreateFolderPathResponse" type="m:CreateFolderPathResponseType"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"/>
```

3.1.4.2.3 Complex Types

The following table summarizes the XML schema complex type definitions that are specific to this operation.

Complex type	Description
CreateFolderPathType	Specifies a request to create a folder path.
CreateFolderPathResponseType	Specifies the response for the CreateFolderPath operation.

3.1.4.2.3.1 CreateFolderPathType

Namespace: http://schemas.microsoft.com/exchange/services/2006/messages

The **CreateFolderPathType** complex type specifies a request to create a folder path.

```
<xs:complexType name="CreateFolderPathType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexContent mixed="false">
    <xs:extension base="m:BaseRequestType">
      <xs:sequence>
        <xs:element name="ParentFolderId" type="t:TargetFolderIdType"/>
        <xs:element name="RelativeFolderPath" type="t:NonEmptyArrayOfFoldersType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

ParentFolderId: Specifies the parent folder identifier.

RelativeFolderPath: Specifies the relative folder path.

3.1.4.2.3.2 CreateFolderPathResponseType

Namespace: http://schemas.microsoft.com/exchange/services/2006/messages

The **CreateFolderPathResponseType** complex type specifies the response for the **CreateFolderPath** operation.

```
<xs:complexType name="CreateFolderPathResponseType"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexContent mixed="false">
    <xs:extension base="m:BaseResponseMessageType"/>
  </xs:complexContent>
</xs:complexType>
```

3.1.4.2.4 Simple Types

None.

3.1.4.2.5 Attributes

None.

3.1.4.2.6 Groups

None.

3.1.4.2.7 Attribute Groups

None.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

4 Protocol Examples

Preliminary

5 Security

5.1 Security Considerations for Implementers

This protocol relies on the web server that hosts the application to perform authentication. The protocol SHOULD use secure communications by means of HTTPS, as described in [\[RFC2818\]](#).

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

For ease of implementation, the full WSDL is provided in this appendix.

```
<?xml version="1.0"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types/>
  <wsdl:message name="ArchiveItemSoapIn">
    <wsdl:part name="request" element="m:ArchiveItem"/>
    <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="ArchiveItemSoapOut">
    <wsdl:part name="ArchiveItemResult" element="m:ArchiveItemResponse"/>
    <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
  </wsdl:message>
  <wsdl:message name="CreateFolderPathSoapIn">
    <wsdl:part name="request" element="m:CreateFolderPath"/>
    <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="CreateFolderPathSoapOut">
    <wsdl:part name="CreateFolderPathResult" element="m:CreateFolderPathResponse"/>
    <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
  </wsdl:message>
  <wsdl:portType name="ExchangeServicePortType">
    <wsdl:operation name="ArchiveItem">
      <wsdl:input message="m:ArchiveItemSoapIn"/>
      <wsdl:output message="m:ArchiveItemSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="CreateFolderPath">
      <wsdl:input message="m:CreateFolderPathSoapIn"/>
      <wsdl:output message="m:CreateFolderPathSoapOut"/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="ExchangeServiceBinding" type="m: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 transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="ArchiveItem">
      <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/ArchiveItem"/>
      <wsdl:input>
        <soap:header message="m:ArchiveItemSoapIn" part="Impersonation" use="literal"/>
        <soap:header message="m:ArchiveItemSoapIn" part="MailboxCulture" use="literal"/>
        <soap:header message="m:ArchiveItemSoapIn" part="RequestVersion" use="literal"/>
        <soap:body use="literal" parts="request"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" parts="ArchiveItemResult"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>

```

```
        <soap:header message="m:ArchiveItemSoapOut" part="ServerVersion" use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CreateFolderPath">
    <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/CreateFolderPath"/>
    <wsdl:input>
        <soap:header message="m:CreateFolderPathSoapIn" part="Impersonation" use="literal"/>
        <soap:header message="m:CreateFolderPathSoapIn" part="MailboxCulture" use="literal"/>
        <soap:header message="m:CreateFolderPathSoapIn" part="RequestVersion" use="literal"/>
        <soap:header message="m:CreateFolderPathSoapIn" part="TimeZoneContext"
use="literal"/>
        <soap:body use="literal" parts="request"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" parts="CreateFolderPathResult"/>
        <soap:header message="m:CreateFolderPathSoapOut" part="ServerVersion" use="literal"/>
    </wsdl:output>
</wsdl:operation>
</wsdl:binding>
</wsdl:definitions>
```

7 Appendix B: Full XML Schema

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

```
<?xml version="1.0"?>
<xs:schema elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
version="Exchange2012" id="messages" xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
  <xs:import namespace="http://schemas.microsoft.com/exchange/services/2006/types"/>
  <xs:include schemaLocation="MS-OXWSARCH-messages.xsd"/>
  <xs:complexType name="ArchiveItemType">
    <xs:complexContent mixed="false">
      <xs:extension base="m:BaseRequestType">
        <xs:sequence>
          <xs:element name="ArchiveSourceFolderId" type="t:TargetFolderIdType"/>
          <xs:element name="ItemIds" type="t:NonEmptyArrayOfBaseItemIdsType"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="ArchiveItem" type="m:ArchiveItemType"/>
  <xs:complexType name="ArchiveItemResponseType">
    <xs:complexContent mixed="false">
      <xs:extension base="m:BaseResponseMessageType"/>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="ArchiveItemResponse" type="m:ArchiveItemResponseType"/>
  <xs:complexType name="CreateFolderPathType">
    <xs:complexContent mixed="false">
      <xs:extension base="m:BaseRequestType">
        <xs:sequence>
          <xs:element name="ParentFolderId" type="t:TargetFolderIdType"/>
          <xs:element name="RelativeFolderPath" type="t:NonEmptyArrayOfFoldersType"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="CreateFolderPath" type="m:CreateFolderPathType"/>
  <xs:complexType name="CreateFolderPathResponseType">
    <xs:complexContent mixed="false">
      <xs:extension base="m:BaseResponseMessageType"/>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="CreateFolderPathResponse" type="m:CreateFolderPathResponseType"/>
</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 2013 Preview
- Microsoft® Outlook® 2013 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.

[<1> Section 3.1.4.1.1.1:](#) Exchange 2007, Exchange 2010, and Exchange 2010 SP1 do not use the **Impersonation** header. The **Impersonation** header was introduced in Exchange 2010 SP2.

[<2> Section 3.1.4.1.1.1:](#) Exchange 2007, Exchange 2010, and Exchange 2010 SP1 do not use the **RequestVersion** header. The **RequestVersion** header was introduced in Exchange 2010 SP2.

[<3> Section 3.1.4.2.1.1:](#) Exchange 2007, Exchange 2010, and Exchange 2010 SP1 do not use the **Impersonation** header. The **Impersonation** header was introduced in Exchange 2010 SP2.

[<4> Section 3.1.4.2.1.1:](#) Exchange 2007, Exchange 2010, and Exchange 2010 SP1 do not use the **RequestVersion** header. The **RequestVersion** header was introduced in Exchange 2010 SP2.

[<5> Section 3.1.4.2.1.1:](#) Exchange 2007, Exchange 2010, and Exchange 2010 SP1 do not use the **TimeZoneContext** header. The **TimeZoneContext** header was introduced in Exchange 2010 SP2.

9 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

Preliminary

10 Index

A

Abstract data model
 [server](#) 11
[Applicability](#) 7
[Attribute groups](#) 10
[Attributes](#) 10

C

[Capability negotiation](#) 7
[Change tracking](#) 26
[Complex types](#) 9

D

Data model - abstract
 [server](#) 11

E

Events
 [local - server](#) 19
 [timer - server](#) 19

F

[Fields - vendor-extensible](#) 7
[Full WSDL](#) 22
[Full XML Schema](#) 24

G

[Glossary](#) 5
[Groups](#) 10

I

[Implementer - security considerations](#) 21
[Index of security parameters](#) 21
[Informative references](#) 6
Initialization
 [server](#) 11
[Introduction](#) 5

L

Local events
 [server](#) 19

M

Message processing
 [server](#) 11
Messages
 [attribute groups](#) 10
 [attributes](#) 10
 [complex types](#) 9
 [elements](#) 9

[enumerated](#) 9
 [groups](#) 10
 [namespaces](#) 9
 [simple types](#) 10
 [syntax](#) 9
 [transport](#) 9

N

[Namespaces](#) 9
[Normative references](#) 5

O

Operations
 [ArchiveItem](#) 11
 [CreateFolderPath](#) 15
 [Overview \(synopsis\)](#) 6

P

[Parameters - security index](#) 21
[Preconditions](#) 7
[Prerequisites](#) 7
[Product behavior](#) 25

R

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

S

Security
 [implementer considerations](#) 21
 [parameter index](#) 21
Sequencing rules
 [server](#) 11
Server
 [abstract data model](#) 11
 [ArchiveItem operation](#) 11
 [CreateFolderPath operation](#) 15
 [initialization](#) 11
 [local events](#) 19
 [message processing](#) 11
 [sequencing rules](#) 11
 [timer events](#) 19
 [timers](#) 11
 [Simple types](#) 10
 [Standards assignments](#) 8
Syntax
 [messages - overview](#) 9

T

Timer events
 [server](#) 19

Timers
 [server](#) 11
[Tracking changes](#) 26
[Transport](#) 9
Types
 [complex](#) 9
 [simple](#) 10

V

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

W

[WSDL](#) 22

X

[XML Schema](#) 24