Delegate Access Management Web Service Protocol

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation ("this documentation") for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.

- **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 can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.

- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.

- **Patents.** Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft Open Specifications Promise or the Microsoft Community Promise. If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.

- **License Programs.** To see all of the protocols in scope under a specific license program and the associated patents, visit the Patent Map.

- **Trademarks.** The names of companies and products contained in this documentation might 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, email addresses, logos, people, places, and events that are 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 as specifically described above, whether by implication, estoppel, or otherwise.

**Tools.** The Open Specifications documentation does 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 documents are intended for use in conjunction with publicly available standards specifications and network programming art and, as such, assume that the reader either is familiar with the aforementioned material or has immediate access to it.

**Support.** For questions and support, please contact dochelp@microsoft.com.
## Revision Summary

<table>
<thead>
<tr>
<th>Date</th>
<th>Revision History</th>
<th>Revision Class</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>7/15/2009</td>
<td>1.0</td>
<td>Major</td>
<td>Initial Availability.</td>
</tr>
<tr>
<td>11/4/2009</td>
<td>1.1.0</td>
<td>Minor</td>
<td>Updated the technical content.</td>
</tr>
<tr>
<td>2/10/2010</td>
<td>1.2.0</td>
<td>Minor</td>
<td>Updated the technical content.</td>
</tr>
<tr>
<td>5/5/2010</td>
<td>1.2.1</td>
<td>Editorial</td>
<td>Revised and edited the technical content.</td>
</tr>
<tr>
<td>8/4/2010</td>
<td>2.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>11/3/2010</td>
<td>3.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>3/18/2011</td>
<td>4.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>8/5/2011</td>
<td>4.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>10/7/2011</td>
<td>4.1</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>1/20/2012</td>
<td>5.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>4/27/2012</td>
<td>5.0</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>7/16/2012</td>
<td>5.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>10/8/2012</td>
<td>5.2</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>2/11/2013</td>
<td>5.2</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>7/26/2013</td>
<td>5.2</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>11/18/2013</td>
<td>5.2</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>2/10/2014</td>
<td>5.2</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>4/30/2014</td>
<td>5.3</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>7/31/2014</td>
<td>5.4</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>10/30/2014</td>
<td>5.4</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>5/26/2015</td>
<td>5.4</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>9/14/2015</td>
<td>5.4</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>6/13/2016</td>
<td>5.4</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>9/14/2016</td>
<td>5.4</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>12/15/2016</td>
<td>5.5</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>Date</td>
<td>Revision History</td>
<td>Revision Class</td>
<td>Comments</td>
</tr>
<tr>
<td>------------</td>
<td>------------------</td>
<td>----------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>7/24/2018</td>
<td>6.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>10/1/2018</td>
<td>7.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
</tbody>
</table>
# Table of Contents

1  Introduction ........................................................................................................... 7
   1.1  Glossary ............................................................................................................. 7
   1.2  References ........................................................................................................ 8
   1.2.1  Normative References ................................................................................ 9
   1.2.2  Informative References .............................................................................. 9
   1.3  Overview ........................................................................................................... 9
   1.4  Relationship to Other Protocols ................................................................. 10
   1.5  Prerequisites/Preconditions ............................................................................ 10
   1.6  Applicability Statement .................................................................................. 10
   1.7  Versioning and Capability Negotiation ....................................................... 10
   1.8  Vendor-Extensible Fields .............................................................................. 11
   1.9  Standards Assignments................................................................................... 11

2  Messages ................................................................................................................ 12
   2.1  Transport .......................................................................................................... 12
   2.2  Common Message Syntax ............................................................................. 12
   2.2.1  Namespaces ................................................................................................ 12
   2.2.2  Messages ...................................................................................................... 12
   2.2.3  Elements ...................................................................................................... 12
   2.2.4  Complex Types ........................................................................................... 13
         2.2.4.1  m:ArrayOfDelegateUserResponseMessageType Complex Type .......... 13
         2.2.4.2  m:BaseDelegateResponseMessageType Complex Type ................... 14
         2.2.4.3  m:BaseDelegateType Complex Type .............................................. 14
         2.2.4.4  m:DelegateUserResponseMessageType Complex Type ............... 15
         2.2.4.5  t:ArrayOfDelegateUserType Complex Type .................................... 15
         2.2.4.6  t:ArrayOfUserIdType Complex Type ............................................ 16
         2.2.4.7  t:DelegatePermissionsType Complex Type .................................... 16
         2.2.4.8  t:DelegateUserType Complex Type ............................................... 17
   2.2.5  Simple Types ............................................................................................... 18
         2.2.5.1  t:DelegateFolderPermissionLevelType Simple Type ...................... 18
         2.2.5.2  t:DeliverMeetingRequestsType Simple Type .................................. 19
   2.2.6  Attributes .................................................................................................... 20
   2.2.7  Groups ......................................................................................................... 20
   2.2.8  Attribute Groups......................................................................................... 20

3  Protocol Details ..................................................................................................... 21
   3.1  ExchangeServicePortType Server Details ................................................. 21
      3.1.1  Abstract Data Model ............................................................................... 21
      3.1.2  Timers ...................................................................................................... 21
      3.1.3  Initialization ............................................................................................. 21
      3.1.4  Message Processing Events and Sequencing Rules ............................. 21
         3.1.4.1  AddDelegate Operation ................................................................. 22
            3.1.4.1.1  Messages ................................................................................. 22
               3.1.4.1.1.1  tns:AddDelegateSoapIn Message ...................................... 22
               3.1.4.1.1.2  tns:AddDelegateSoapOut Message ................................... 23
            3.1.4.1.2  Elements .................................................................................... 24
               3.1.4.1.2.1  AddDelegate Element ......................................................... 24
               3.1.4.1.2.2  AddDelegateResponse Element ......................................... 24
            3.1.4.1.3  Complex Types ........................................................................... 24
               3.1.4.1.3.1  m:AddDelegateResponseMessageType Complex Type ........ 25
               3.1.4.1.3.2  m:AddDelegateType Complex Type .................................. 25
            3.1.4.1.4  Simple Types ............................................................................... 25
            3.1.4.1.5  Attributes .................................................................................... 26
            3.1.4.1.6  Groups ........................................................................................ 26
         3.1.4.2  GetDelegate Operation ................................................................. 26
3.1.4.2.1 Messages ................................................................. 26
3.1.4.2.1.1 tns:GetDelegateSoapIn Message ..................................... 26
3.1.4.2.1.2 tns:GetDelegateSoapOut Message ................................. 27
3.1.4.2.2 Elements ............................................................... 28
3.1.4.2.2.1 GetDelegate Element .............................................. 28
3.1.4.2.2.2 GetDelegateResponse Element .................................. 28
3.1.4.2.3 Complex Types ....................................................... 28
3.1.4.2.3.1 m:GetDelegateResponseMessageType Complex Type ............ 28
3.1.4.2.3.2 m:GetDelegateType Complex Type .............................. 29
3.1.4.2.4 Simple Types ......................................................... 30
3.1.4.2.5 Attributes ............................................................ 30
3.1.4.2.6 Groups ............................................................... 30
3.1.4.3 RemoveDelegate Operation .......................................... 30
3.1.4.3.1 Messages ............................................................. 30
3.1.4.3.1.1 tns:RemoveDelegateSoapIn Message ............................ 31
3.1.4.3.1.2 tns:RemoveDelegateSoapOut Message .......................... 31
3.1.4.3.2 Elements ............................................................. 32
3.1.4.3.2.1 RemoveDelegateResponse Element .............................. 32
3.1.4.3.2.2 RemoveDelegate Element ........................................ 32
3.1.4.3.3 Complex Types ....................................................... 32
3.1.4.3.3.1 m:RemoveDelegateType Complex Type .......................... 33
3.1.4.3.3.2 m:RemoveDelegateResponseMessageType Complex Type ....... 33
3.1.4.3.4 Simple Types ......................................................... 33
3.1.4.3.5 Attributes ............................................................ 34
3.1.4.3.6 Groups ............................................................... 34
3.1.4.4 UpdateDelegate Operation ........................................... 34
3.1.4.4.1 Messages ............................................................. 34
3.1.4.4.1.1 tns:UpdateDelegateSoapIn Message ............................ 34
3.1.4.4.1.2 tns:UpdateDelegateSoapOut Message .......................... 35
3.1.4.4.2 Elements ............................................................. 36
3.1.4.4.2.1 UpdateDelegateResponse Element .............................. 36
3.1.4.4.2.2 UpdateDelegate Element ........................................ 36
3.1.4.4.3 Complex Types ....................................................... 36
3.1.4.4.3.1 m:UpdateDelegateType Complex Type .......................... 37
3.1.4.4.3.2 m:UpdateDelegateResponseMessageType Complex Type ....... 37
3.1.4.4.4 Simple Types ......................................................... 38
3.1.4.4.5 Attributes ............................................................ 38
3.1.4.4.6 Groups ............................................................... 38
3.1.5 Timer Events ............................................................. 38
3.1.6 Other Local Events ....................................................... 38

4 Protocol Examples .................................................................. 39
4.1 AddDelegate Operation ...................................................... 39
4.2 GetDelegate Operation ...................................................... 40
4.3 RemoveDelegate Operation ............................................... 41
4.4 UpdateDelegate Operation ............................................... 42

5 Security .............................................................................. 44
5.1 Security Considerations for Implementers ............................... 44
5.2 Index of Security Parameters .............................................. 44

6 Appendix A: Full WSDL ........................................................ 45

7 Appendix B: Full XML Schema ............................................. 48
7.1 Messages Schema ............................................................ 48
7.2 Types Schema .................................................................. 50

8 Appendix C: Product Behavior ................................................. 52

9 Change Tracking .................................................................. 53
1 Introduction

The Delegate Access Management Web Service Protocol is used to manage delegate access to mailbox and calendar information that is stored on a server.

Sections 1.5, 1.8, 1.9, 2, and 3 of this specification are normative. All other sections and examples in this specification are informative.

1.1 Glossary

This document uses the following terms:

- **Calendar folder**: A Folder object that contains Calendar objects.
- **Contacts folder**: A Folder object that contains Contact objects.
- **delegate**: A user or resource that has permissions to act on behalf of another user or resource.
- **delegate access**: The access that is granted by a delegator to a delegate and is used by the delegate to access the delegator's account.
- **Deleted Items folder**: A special folder that is the default location for objects that have been deleted.
- **endpoint**: A communication port that is exposed by an application server for a specific shared service and to which messages can be addressed.
- **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 Secure (HTTPS)**: An extension of HTTP that securely encrypts and decrypts web page requests. In some older protocols, "Hypertext Transfer Protocol over Secure Sockets Layer" is still used (Secure Sockets Layer has been deprecated). For more information, see [SSL3] and [RFC5246].
- **Inbox folder**: A special folder that is the default location for Message objects received by a user or resource.
- **Journal folder**: A Folder object that contains Journal objects.
- **mailbox**: A message store that contains email, calendar items, and other Message objects for a single recipient.
- **meeting request**: An instance of a Meeting Request object.
- **Notes folder**: A Folder object that contains Note objects.
- **permission**: A rule that is associated with an object and that regulates which users can gain access to the object and in what manner. See also rights.
- **security identifier (SID)**: An identifier for security principals that is used to identify an account or a group. Conceptually, the SID is composed of an account authority portion (typically a domain) and a smaller integer representing an identity relative to the account authority, termed the relative identifier (RID). The SID format is specified in [MS-DTYP] section 2.4.2; a string representation of SIDs is specified in [MS-DTYP] section 2.4.2 and [MS-AZOD] section 1.1.1.2.
- **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].

**SOAP action**: The HTTP request header field used to indicate the intent of the SOAP request, using a URI value. See [SOAP1.1] section 6.1.1 for more information.

**SOAP body**: A container for the payload data being delivered by a SOAP message to its recipient. See [SOAP1.2-1/2007] section 5.3 for more information.

**SOAP header**: A mechanism for implementing extensions to a SOAP message in a decentralized manner without prior agreement between the communicating parties. See [SOAP1.2-1/2007] section 5.2 for more information.

**SOAP message**: An XML document consisting of a mandatory SOAP envelope, an optional SOAP header, and a mandatory SOAP body. See [SOAP1.2-1/2007] section 5 for more information.

**Tasks folder**: A Folder object that contains Task objects.

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


1.2.2 Informative References


1.3 Overview

Delegates are mailbox owners who are authorized to act on behalf of other mailbox owners. Client applications can use the Delegate Access Management Web Service Protocol to add, update, and remove delegates, and to query the server to determine whether a specific delegate is authorized to act on a specific mailbox.
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 server. This protocol uses the XML Protocol, as described in [XMLSCHEMA1] and [XMLSCHEMA2], to describe the message content sent to and from the server.

This protocol uses SOAP over HTTP, as described in [RFC2616], and SOAP over HTTPS, as described in [RFC2818], as shown in the following figure.

![Figure 1: This protocol in relation to other protocols](image)

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 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 is identified in the HTTP Web requests that target this protocol.

1.6 Applicability Statement

This protocol is applicable to client applications that manage mailbox delegates on a server.

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

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

- **Protocol Versions:** This protocol has only one WSDL port type version. The WSDL version of the request is identified by using the `t:RequestServerVersion` element, as described in [MS-OXWSCDATA] section 2.2.3.9, and the version of the server responding to the request is identified by using the `t:ServerVersionInfo` element, as described in [MS-OXWSCDATA] section 2.2.3.10.

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

- **Localization:** This protocol includes text strings in various messages. Localization considerations for such strings are specified in sections 2.2 and 3.1.4.
**Capability Negotiation:** This protocol does not support version negotiation.

### 1.8 Vendor-Extensible Fields

None.

### 1.9 Standards Assignments

None.
2  Messages

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

2.1  Transport

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

This protocol relies on the web server that hosts the application to perform authentication. The protocol MUST support SOAP over HTTP, as specified in [RFC2616]. The protocol SHOULD use secure communications via HTTPS, as defined in [RFC2818].

2.2  Common Message Syntax

This section contains common definitions that are used by this protocol. The syntax of the definitions uses **XML schema**, as defined in [XMLSCHEMA1] and [XMLSCHEMA2], and **WSDL**, as defined 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.

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Namespace URI</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>soap</td>
<td><a href="http://schemas.xmlsoap.org/wsdl/soap/">http://schemas.xmlsoap.org/wsdl/soap/</a></td>
<td>[SOAP1.1]</td>
</tr>
<tr>
<td>tns</td>
<td><a href="http://schemas.microsoft.com/exchange/services/2006/messages">http://schemas.microsoft.com/exchange/services/2006/messages</a></td>
<td></td>
</tr>
<tr>
<td>wsdl</td>
<td><a href="http://schemas.xmlsoap.org/wsdl/">http://schemas.xmlsoap.org/wsdl/</a></td>
<td>[WSDL]</td>
</tr>
<tr>
<td>t</td>
<td><a href="http://schemas.microsoft.com/exchange/services/2006/types">http://schemas.microsoft.com/exchange/services/2006/types</a></td>
<td></td>
</tr>
<tr>
<td>m</td>
<td><a href="http://schemas.microsoft.com/exchange/services/2006/messages">http://schemas.microsoft.com/exchange/services/2006/messages</a></td>
<td></td>
</tr>
<tr>
<td>xs</td>
<td><a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a></td>
<td>[XMLSCHEMA1]</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Complex type name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArrayOfDelegateUserResponseMessageType</td>
<td>Specifies one or more delegates for a mailbox.</td>
</tr>
<tr>
<td>BaseDelegateResponseMessageType</td>
<td>Specifies the base type for all Delegate Access Management Web Service Protocol response types.</td>
</tr>
<tr>
<td>BaseDelegateType</td>
<td>Specifies the base type for derived types that specify all delegate management requests.</td>
</tr>
<tr>
<td>DelegateUserResponseMessageType</td>
<td>Specifies a response message for a single delegate.</td>
</tr>
<tr>
<td>ArrayOfDelegateUserType</td>
<td>Specifies one or more responses from a Delegate Access Management Web Service Protocol operation.</td>
</tr>
<tr>
<td>ArrayOfUserIdType</td>
<td>Specifies one or more user identifiers.</td>
</tr>
<tr>
<td>DelegatePermissionsType</td>
<td>Specifies the delegate permissions for a mailbox delegate.</td>
</tr>
<tr>
<td>Delegate UserType</td>
<td>Specifies a single delegate to add or update in a mailbox, or a delegate that is returned in a delegate management response.</td>
</tr>
</tbody>
</table>

2.2.4.1 m:ArrayOfDelegateUserResponseMessageType Complex Type

The ArrayOfDelegateUserResponseMessageType complex type specifies one or more responses from a Delegate Access Management Web Service Protocol operation.

```xml
<xs:complexType name="ArrayOfDelegateUserResponseMessageType">
  <xs:sequence>
    <xs:element name="DelegateUserResponseMessageType" type="m:DelegateUserResponseMessageType" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

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

<table>
<thead>
<tr>
<th>Element name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DelegateUserResponseMessageType</td>
<td>m:DelegateUserResponseMessageType (section 2.2.4.4)</td>
<td>Specifies a response from a Delegate Access Management Web Service Protocol operation.</td>
</tr>
</tbody>
</table>

The ArrayOfDelegateUserResponseMessageType complex type is used in the BaseDelegateResponseMessageType complex type, as specified in section 2.2.4.2.
2.2.4.2 m:BaseDelegateResponseMessageType Complex Type

The **BaseDelegateResponseMessageType** complex type is the base type for all Delegate Access Management Web Service Protocol response types. The **BaseDelegateResponseMessageType** complex type extends the **ResponseMessageType** complex type, as specified in [MS-OXWSCDATA] section 2.2.4.67.

```xml
<xs:complexType name="BaseDelegateResponseMessageType"
                      abstract="true">
  <xs:complexContent>
    <xs:extension base="m:ResponseMessageType">
      <xs:sequence>
        <xs:element name="ResponseMessages"
                     type="m:ArrayOfDelegateUserResponseMessageType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

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

<table>
<thead>
<tr>
<th>Element name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseMessages</td>
<td>m:ArrayOfDelegateUserResponseMessageType</td>
<td>Specifies one or more response messages from a Delegate Access Management Web Service Protocol operation. Can be present.</td>
</tr>
</tbody>
</table>

2.2.4.3 m:BaseDelegateType Complex Type

The **BaseDelegateType** complex type specifies the base type for derived types that specify all delegate management requests. The **BaseDelegateType** complex type extends the **BaseRequestType** complex type, as specified in [MS-OXWSCDATA] section 2.2.4.17.

```xml
<xs:complexType name="BaseDelegateType"
                      abstract="true">
  <xs:complexContent>
    <xs:extension base="m:BaseRequestType">
      <xs:sequence>
        <xs:element name="Mailbox"
                     type="t:EmailAddressType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The following table lists the child elements of the **BaseDelegateType** complex type.
### Element name | Type | Description
---|---|---
Mailbox | t:EmailAddressType ([MS-OXWSCDATA] section 2.2.4.31) | Specifies an e-mail address that defines the target of the delegate management operation.

**2.2.4.4 m:DelegateUserResponseMessageType Complex Type**

The `DelegateUserResponseMessageType` complex type specifies a response message for a single delegate. The `DelegateUserResponseMessageType` complex type extends the `ResponseMessageType` complex type, as specified in [MS-OXWSCDATA] section 2.2.4.67.

```xml
<x:s:complexType name="DelegateUserResponseMessageType">
  <xs:complexContent>
    <xs:extension base="m:ResponseMessageType">
      <xs:sequence>
        <xs:element name="DelegateUser"
          type="t:DelegateUserType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The following table lists the child elements of the `DelegateUserResponseMessageType` complex type.

<table>
<thead>
<tr>
<th>Element name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DelegateUser</td>
<td>t:DelegateUserType (section 2.2.4.8)</td>
<td>Specifies a single delegate that is associated with a mailbox that is returned by a delegate management operation.</td>
</tr>
</tbody>
</table>

**2.2.4.5 t:ArrayOfDelegateUserType Complex Type**

The `ArrayOfDelegateUserType` complex type specifies one or more delegates for a mailbox.

```xml
<x:s:complexType name="ArrayOfDelegateUserType">
  <xs:sequence>
    <xs:element name="DelegateUser"
      type="t:DelegateUserType"
      maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

The following table lists the child elements of the `ArrayOfDelegateUserType` complex type.

<table>
<thead>
<tr>
<th>Element name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DelegateUser</td>
<td>t:DelegateUserType (section 2.2.4.8)</td>
<td>Specifies one or more mailbox delegates.</td>
</tr>
</tbody>
</table>
2.2.4.6 t:ArrayOfUserIdType Complex Type

The **ArrayOfUserIdType** complex type specifies one or more user identifiers.

```xml
<xs:complexType name="ArrayOfUserIdType">
  <xs:sequence>
    <xs:element name="UserId" type="t:UserIdType" maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>
```

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

<table>
<thead>
<tr>
<th>Element name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UserId</td>
<td>t:UserIdType ([MS-OXWSDC] section 2.2.4.75)</td>
<td>Specifies one user identifier.</td>
</tr>
</tbody>
</table>

2.2.4.7 t:DelegatePermissionsType Complex Type

The **DelegatePermissionsType** complex type specifies the delegate permissions for a mailbox delegate.

```xml
<xs:complexType name="DelegatePermissionsType">
  <xs:sequence>
    <xs:element name="CalendarFolderPermissionLevel" type="t:DelegateFolderPermissionLevelType" minOccurs="0" />
    <xs:element name="TasksFolderPermissionLevel" type="t:DelegateFolderPermissionLevelType" minOccurs="0" />
    <xs:element name="InboxFolderPermissionLevel" type="t:DelegateFolderPermissionLevelType" minOccurs="0" />
    <xs:element name="ContactsFolderPermissionLevel" type="t:DelegateFolderPermissionLevelType" minOccurs="0" />
    <xs:element name="NotesFolderPermissionLevel" type="t:DelegateFolderPermissionLevelType" minOccurs="0" />
    <xs:element name="JournalFolderPermissionLevel" type="t:DelegateFolderPermissionLevelType" minOccurs="0" />
  </xs:sequence>
</xs:complexType>
```

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

[MS-OXWSDLGM] - v20181001
Delegate Access Management Web Service Protocol
Copyright © 2018 Microsoft Corporation
Release: October 1, 2018
### Element name | Type | Description
--- | --- | ---
CalendarFolderPermissionLevel | t:DelegateFolderPermissionLevelType | Specifies the delegate permissions for the default Calendar folder. Can be present.

TasksFolderPermissionLevel | t:DelegateFolderPermissionLevelType | Specifies the delegate permissions for the default Tasks folder. Can be present.

InboxFolderPermissionLevel | t:DelegateFolderPermissionLevelType | Specifies the delegate permissions for the default Inbox folder. Can be present.

ContactsFolderPermissionLevel | t:DelegateFolderPermissionLevelType | Specifies the delegate permissions for the default Contacts folder. Can be present.

NotesFolderPermissionLevel | t:DelegateFolderPermissionLevelType | Specifies the delegate permissions for the default Notes folder. Can be present.

JournalFolderPermissionLevel | t:DelegateFolderPermissionLevelType | Specifies the delegate permissions for the default Journal folder. Can be present.

### 2.2.4.8 t:DelegateUserType Complex Type

The **DelegateUserType** complex type specifies a single delegate to add or update in a mailbox, or a delegate that is returned in a delegate management response.

```xml
<xs:complexType name="DelegateUserType">
<xs:sequence>
  <xs:element name="UserId" type="t:UserIdType" />
  <xs:element name="DelegatePermissions" type="t:DelegatePermissionsType" minOccurs="0" />
  <xs:element name="ReceiveCopiesOfMeetingMessages" type="xs:boolean" minOccurs="0" />
  <xs:element name="ViewPrivateItems" type="xs:boolean" minOccurs="0" />
</xs:sequence>
</xs:complexType>
```

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

### Element name | Type | Description
--- | --- | ---
UserId | t:UserIdType (MS- | Specifies the user identifier for the
<table>
<thead>
<tr>
<th>Element name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DelegatePermissions</td>
<td>t:DelegatePermissionsType</td>
<td>Specifies the delegate permissions for the delegate user. Can be present.</td>
</tr>
<tr>
<td>ReceiveCopiesOfMeetingMessages</td>
<td>xs:boolean</td>
<td>Specifies a Boolean value that indicates whether the delegate receives copies of meeting-related messages that are addressed to the primary account.</td>
</tr>
<tr>
<td>ViewPrivateItems</td>
<td>xs:boolean</td>
<td>Specifies a Boolean value that indicates whether the delegate can view private calendar items on the primary account's calendar.</td>
</tr>
</tbody>
</table>

### 2.2.5 Simple Types

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

<table>
<thead>
<tr>
<th>Simple type name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DelegateFolderPermissionLevelType</td>
<td>Specifies the permissions that are granted on a folder to a delegate.</td>
</tr>
<tr>
<td>DeliverMeetingRequestType</td>
<td>Specifies whether a delegate receives meeting requests on behalf of a primary account.</td>
</tr>
</tbody>
</table>

#### 2.2.5.1 t:DelegateFolderPermissionLevelType Simple Type

The DelegateFolderPermissionLevelType simple type specifies the permissions that are granted on a folder to a delegate.

```xml
<xs:simpleType name="DelegateFolderPermissionLevelType">  
  <xs:restriction base="xs:string">  
    <xs:enumeration value="None" />  
    <xs:enumeration value="Editor" />  
    <xs:enumeration value="Reviewer" />  
    <xs:enumeration value="Author" />  
    <xs:enumeration value="Custom" />  
  </xs:restriction>  
</xs:simpleType>
```
The following table lists the values that are defined by the DelegateFolderPermissionLevelType simple type.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>Delegate has no permissions on the folder.</td>
</tr>
<tr>
<td>Editor</td>
<td>Delegate can read, create, and modify items in the folder.</td>
</tr>
<tr>
<td>Reviewer</td>
<td>Delegate can read items in the folder.</td>
</tr>
<tr>
<td>Author</td>
<td>Delegate can read and create items in the folder.</td>
</tr>
<tr>
<td>Custom</td>
<td>This setting is not applicable to this protocol.</td>
</tr>
</tbody>
</table>

### 2.2.5.2 DeliverMeetingRequestsType Simple Type

The DeliverMeetingRequestsType simple type specifies how meeting requests are handled between the delegate and the primary account.

```
<xs:simpleType name="DeliverMeetingRequestsType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="DelegatesOnly"/>
    <xs:enumeration value="DelegatesAndMe"/>
    <xs:enumeration value="DelegatesAndSendInformationToMe"/>
    <xs:enumeration value="NoForward"/>
  </xs:restriction>
</xs:simpleType>
```

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

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DelegatesOnly</td>
<td>Specifies that meeting requests are forwarded to the delegate and moved to the Deleted Items folder in the primary account's mailbox.</td>
</tr>
<tr>
<td>DelegatesAndMe</td>
<td>Specifies that meeting requests are forwarded to the delegate and remain in the Inbox folder in the primary account's mailbox.</td>
</tr>
<tr>
<td>DelegatesAndSendInformationToMe</td>
<td>Specifies that meeting requests are forwarded to the delegate and remain in the Inbox folder in the primary account's mailbox.</td>
</tr>
<tr>
<td>NoForward</td>
<td>Specifies that meeting requests are not to be forwarded to delegates.</td>
</tr>
</tbody>
</table>
2.2.6 Attributes
This specification does not define any common XML schema attribute definitions.

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

2.2.8 Attribute Groups
This specification does not define any common XML schema attribute group definitions.
3 Protocol Details

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

3.1 ExchangeServicePortType Server Details

The Delegate Access Management Web Service Protocol defines a single port type with four operations. The operations enable client implementations to create, review, update, and delete delegates for a primary account.

3.1.1 Abstract Data Model

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

The Delegate Access Management Web Service Protocol is used to set, retrieve, update, and delete delegate access permissions on a primary account's mailbox on the server. The server maintains the delegate access permissions and modifies them as requested.

The client is not required to maintain the state of delegate access permissions on the server and can retrieve the current set of permissions at any time. If more than one client is changing delegate access permissions, there is no requirement that the server lock the existing set of permissions for changes.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing Rules

This protocol includes the operations that are listed in the following table.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddDelegate</td>
<td>Adds one or more delegates to a primary account's mailbox.</td>
</tr>
<tr>
<td>GetDelegate</td>
<td>Gets the delegates that are associated with a primary account's mailbox, if any.</td>
</tr>
<tr>
<td>RemoveDelegate</td>
<td>Removes one or more delegates from the primary account's mailbox.</td>
</tr>
<tr>
<td>UpdateDelegate</td>
<td>Modifies one or more delegates.</td>
</tr>
</tbody>
</table>
### 3.1.4.1 AddDelegate Operation

The **AddDelegate** operation adds one or more **delegates** to a **mailbox**.

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

```xml
<wsdl:operation name="AddDelegate">
    <wsdl:input message="tns:AddDelegateSoapIn" />
    <wsdl:output message="tns:AddDelegateSoapOut" />
</wsdl:operation>
```

The following is the WSDL binding specification of the operation.

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

### 3.1.4.1.1 Messages

The following table lists the **WSDL message** definitions that are specific to this operation.

<table>
<thead>
<tr>
<th>Message name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddDelegateSoapIn</td>
<td>Specifies the <strong>SOAP message</strong> that adds one or more <strong>delegates</strong> to the <strong>mailbox</strong>.</td>
</tr>
<tr>
<td>AddDelegateSoapOut</td>
<td>Specifies the SOAP message that is returned by the server in response.</td>
</tr>
</tbody>
</table>

#### 3.1.4.1.1.1 tns:AddDelegateSoapIn Message

The **AddDelegateSoapIn** **WSDL message** specifies the **AddDelegate** operation request to add one or more **delegates** to a **mailbox**.

```xml
<wsdl:message name="AddDelegateSoapIn">
    <wsdl:part name="request" element="tns:AddDelegate" />
    <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
    <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
    <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
</wsdl:message>
```

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

The parts of the **AddDelegateSoapIn** WSDL message are described in the following table.
### 3.1.4.1.2 WSDL Elements

#### request

<table>
<thead>
<tr>
<th>Part name</th>
<th>Element/type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>request</td>
<td>tns:AddDelegate (section 3.1.4.1.2.1)</td>
<td>Specifies the SOAP body of the request to add one or more delegates to a mailbox.</td>
</tr>
</tbody>
</table>

| Impersonation | t:ExchangeImpersonation ([MS-OXWSCDATA] section 2.2.3.3) | Specifies a SOAP header that identifies the user whom the client application is impersonating. |

| MailboxCulture | t:MailboxCulture ([MS-OXWSCDATA] section 2.2.3.6) | Specifies a SOAP header that identifies the culture to use for accessing the mailbox. The cultures are defined by [RFC3066]. |

| RequestVersion | t:RequestServerVersion ([MS-OXWSCDATA] section 2.2.3.9) | Specifies a SOAP header that identifies the schema version for the AddDelegate operation request. |

#### 3.1.4.1.2.2 WSDL Message

The AddDelegate WSDL message specifies the server response to the AddDelegate operation request to add one or more delegates to a mailbox.

```xml
<wsdl:message name="AddDelegateSoapOut">
  <wsdl:part name="AddDelegateResult" element="tns:AddDelegateResponse" />
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
```

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

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

<table>
<thead>
<tr>
<th>Part name</th>
<th>Element/type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddDelegateResult</td>
<td>tns:AddDelegateResponse (section 3.1.4.1.2.2)</td>
<td>Specifies the SOAP body of the response to an AddDelegate operation request.</td>
</tr>
<tr>
<td>ServerVersion</td>
<td>t:ServerVersionInfo ([MS-OXWSCDATA] section 2.2.3.10)</td>
<td>Specifies a SOAP header that identifies the server version for the response.</td>
</tr>
</tbody>
</table>

The AddDelegate operation returns an AddDelegateResponse element with the ResponseClass attribute set to "Success". The ResponseCode element of the AddDelegateResponse element is set to "NoError".

The operation returns the ResponseMessages element of the AddDelegateResponse element. A DelegateUserResponseMessageType element of the ResponseMessages element is returned for each DelegateUser element in the AddDelegate request. If the delegate was added successfully, the ResponseClass attribute of the DelegateUserResponseMessageType element is set to "Success". The ResponseCode element of the DelegateUserResponseMessageType element is set to "NoError".

Each successful DelegateUserResponseMessageType element includes a DelegateUser element, which includes a UserId element, a ReceiveCopiesOfMeetingRequests element, and a ViewPrivateItems element.

If the delegate was not added successfully, the ResponseClass attribute of the DelegateUserResponseMessageType element is set to "Error". The ResponseCode element of the
DelegateUserResponseMessageType can be one of the error values listed in the following table. These errors and other common errors are defined in [MS-OXWSCDATA] section 2.2.5.24.

<table>
<thead>
<tr>
<th>Error Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ErrorAddDelegatesFailed</td>
<td>Occurs when a list with added delegates cannot be saved.</td>
</tr>
<tr>
<td>ErrorDelegateAlreadyExists</td>
<td>Occurs in an AddDelegate operation when the specified user already exists in the list of delegates.</td>
</tr>
<tr>
<td>ErrorDelegateCannotAddOwner</td>
<td>Occurs in an AddDelegate operation when the specified user to be added is the owner of the mailbox.</td>
</tr>
<tr>
<td>ErrorDelegateValidationFailed</td>
<td>Occurs in an AddDelegate operation when an added delegate user is not valid.</td>
</tr>
<tr>
<td>ErrorInvalidDelegatePermission</td>
<td>Specifies that the specified folder permissions are invalid.</td>
</tr>
</tbody>
</table>

### 3.1.4.1.2 Elements

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

<table>
<thead>
<tr>
<th>Element name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddDelegate</td>
<td>Specifies a request to add one or more delegates to a mailbox.</td>
</tr>
<tr>
<td>AddDelegateResponse</td>
<td>Specifies the response body content from a request to add one or more delegates to a mailbox.</td>
</tr>
</tbody>
</table>

#### 3.1.4.1.2.1 AddDelegate Element

The AddDelegate element specifies the request message for an AddDelegate operation.

```xml
<xs:element name="AddDelegate"
             type="m:AddDelegateType"/>
```

#### 3.1.4.1.2.2 AddDelegateResponse Element

The AddDelegateResponse element specifies the response message for an AddDelegate operation.

```xml
<xs:element name="AddDelegateResponse"
             type="m:AddDelegateResponseMessageType"/>
```

#### 3.1.4.1.3 Complex Types

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

<table>
<thead>
<tr>
<th>Complex type name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddDelegateResponseMessageType</td>
<td>Specifies the response message for the AddDelegate operation.</td>
</tr>
</tbody>
</table>
### Complex Type name | Description
--- | ---
**AddDelegateType** | Specifies the request message for the AddDelegate operation.

#### 3.1.4.1.3.1 m:AddDelegateResponseMessageType Complex Type

The AddDelegateResponseMessageType complex type specifies the response message from the AddDelegate operation. The AddDelegateResponseMessageType complex type extends the BaseDelegateResponseMessageType complex type, as specified in section 2.2.4.2.

```xml
<xs:complexType name="AddDelegateResponseMessageType">
  <xs:complexContent>
    <xs:extension base="m:BaseDelegateResponseMessageType"/>
  </xs:complexContent>
</xs:complexType>
```

#### 3.1.4.1.3.2 m:AddDelegateType Complex Type

The AddDelegateType complex type specifies a request to add one or more delegates to a primary account's mailbox by using the AddDelegate operation. The AddDelegateType complex type extends the BaseDelegateType complex type, as specified in section 2.2.4.3.

```xml
<xs:complexType name="AddDelegateType">
  <xs:complexContent>
    <xs:extension base="m:BaseDelegateType">
      <xs:sequence>
        <xs:element name="DelegateUsers" type="t:ArrayOfDelegateUserType"/>
        <xs:element name="DeliverMeetingRequests" type="t:DeliverMeetingRequestsType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

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

<table>
<thead>
<tr>
<th>Element name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DelegateUsers</td>
<td>t:ArrayOfDelegateUserType</td>
<td>Specifies the delegates to add to the mailbox.</td>
</tr>
<tr>
<td>DeliverMeetingRequests</td>
<td>t:DeliverMeetingRequestsType</td>
<td>Specifies how meeting-related messages are handled between the delegate and the primary account.</td>
</tr>
</tbody>
</table>

#### 3.1.4.1.4 Simple Types
None.

### 3.1.4.1.5 Attributes
None.

### 3.1.4.1.6 Groups
None.

#### 3.1.4.2 GetDelegate Operation

The **GetDelegate** operation gets one or more **delegates** that are assigned to a **mailbox**.

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

```xml
<wSDL:operation name="GetDelegate">
  <wSDL:input message="tns:GetDelegateSoapIn" />
  <wSDL:output message="tns:GetDelegateSoapOut" />
</wSDL:operation>
```

The following is the WSDL binding specification for the operation.

```xml
<wSDL:operation name="GetDelegate">
  <wSDL:input>
    <soap:header message="tns:GetDelegateSoapIn" part="Impersonation" use="literal"/>
    <soap:header message="tns:GetDelegateSoapIn" part="MailboxCulture" use="literal"/>
    <soap:header message="tns:GetDelegateSoapIn" part="RequestVersion" use="literal"/>
    <soap:body parts="request" use="literal" />
  </wSDL:input>
  <wSDL:output>
    <soap:body parts="GetDelegateResult" use="literal" />
    <soap:header message="tns:GetDelegateSoapOut" part="ServerVersion" use="literal"/>
  </wSDL:output>
</wSDL:operation>
```

#### 3.1.4.2.1 Messages

The following table lists the **WSDL message** definitions that are specific to this operation.

<table>
<thead>
<tr>
<th>Message name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetDelegateSoapIn</td>
<td>Specifies the SOAP message that requests one or more delegates to the mailbox.</td>
</tr>
<tr>
<td>GetDelegateSoapOut</td>
<td>Specifies the SOAP message that is returned by the server in response.</td>
</tr>
</tbody>
</table>

#### 3.1.4.2.1.1 tns:GetDelegateSoapIn Message

The **GetDelegateSoapIn** WSDL message specifies the **GetDelegate** operation request to return one or more delegates to a mailbox.

```xml
<wSDL:message name="GetDelegateSoapIn">
  <wSDL:part name="request" element="tns:GetDelegate" />
</wSDL:message>
```
The `GetDelegateSoapIn` WSDL message is the input message for the SOAP action http://schemas.microsoft.com/exchange/services/2006/messages/GetDelegate.

The parts of the `GetDelegateSoapIn` WSDL message are described in the following table.

<table>
<thead>
<tr>
<th>Part name</th>
<th>Element/type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>request</td>
<td>tns:GetDelegate (section 3.1.4.2.2.1)</td>
<td>Specifies the SOAP body of the request to return one or more mailbox delegates.</td>
</tr>
<tr>
<td>Impersonation</td>
<td>t:ExchangeImpersonation ([MS-OXWSCDATA] section 2.2.3.3)</td>
<td>Specifies a SOAP header that identifies the user whom the client application is impersonating.</td>
</tr>
<tr>
<td>MailboxCulture</td>
<td>t:MailboxCulture ([MS-OXWSCDATA] section 2.2.3.6)</td>
<td>Specifies a SOAP header that identifies the culture to use for accessing the mailbox. The cultures are specified by [RFC3066].</td>
</tr>
<tr>
<td>RequestVersion</td>
<td>t:RequestServerVersion ([MS-OXWSCDATA] section 2.2.3.9)</td>
<td>Specifies a SOAP header that identifies the schema version for the <code>GetDelegate</code> operation request.</td>
</tr>
</tbody>
</table>

### 3.1.4.2.1.2 tns:GetDelegateSoapOut Message

The `GetDelegateSoapOut` WSDL message specifies the server response to the `GetDelegate` operation request to return one or more mailbox delegates.

```xml
<wsdl:message name="GetDelegateSoapOut">
  <wsdl:part name="GetDelegateResult" element="tns:GetDelegateResponse" />
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
```

The `GetDelegateSoapOut` WSDL message is the output message for the SOAP action http://schemas.microsoft.com/exchange/services/2006/messages/GetDelegate.

The parts of the `GetDelegateSoapOut` WSDL message are described in the following table.

<table>
<thead>
<tr>
<th>Part name</th>
<th>Element/type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetDelegateResult</td>
<td>tns:GetDelegateResponse (section 3.1.4.2.2)</td>
<td>Specifies the SOAP body of the response to a <code>GetDelegate</code> operation request.</td>
</tr>
<tr>
<td>ServerVersion</td>
<td>t:ServerVersionInfo ([MS-OXWSCDATA] section 2.2.3.10)</td>
<td>Specifies a SOAP header that identifies the server version for the response.</td>
</tr>
</tbody>
</table>

The `GetDelegate` operation returns a `GetDelegateResponse` element with the `ResponseClass` attribute set to "Success". The `ResponseCode` element of the `GetDelegateResponse` element is set to "NoError".

If there are delegates assigned to the mailbox, the operation returns the `ResponseMessages` element of the `GetDelegateResponse` element. A `DelegateUserResponseMessageType` element of the `ResponseMessages` element is returned for each delegate returned. If a delegate was
retrieved successfully, the ResponseClass attribute of the DelegateUserResponseMessageType element is set to "Success". The ResponseCode element of the DelegateUserResponseMessageType is set to "NoError".

Each successful DelegateUserResponseMessageType element includes a UserId element, a DelegatePermissions element, a ReceiveCopiesOfMeetingMessages element, and a ViewPrivateItems element.

If the request was not successful, the ResponseClass attribute of the DelegateUserResponseMessageType element is set to "Error". The ResponseCode element of the DelegateUserResponseMessageType is set to a value of the ResponseCodeType simple type, as specified in [MS-OXWSCDATA] section 2.2.5.24.

3.1.4.2.2 Elements

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

<table>
<thead>
<tr>
<th>Element name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetDelegate</td>
<td>Specifies a request to get one or more delegates to a mailbox.</td>
</tr>
<tr>
<td>GetDelegateResponse</td>
<td>Specifies the response body content from a request to get delegates to a mailbox.</td>
</tr>
</tbody>
</table>

3.1.4.2.2.1 GetDelegate Element

The GetDelegate element specifies the base element for a GetDelegate operation.

```xml
<xs:element name="GetDelegate"
    type="m:GetDelegateType"/>
```

3.1.4.2.2.2 GetDelegateResponse Element

The GetDelegateResponse element specifies the response message for a GetDelegate operation.

```xml
<xs:element name="GetDelegateResponse"
    type="m:GetDelegateResponseMessageType"/>
```

3.1.4.2.3 Complex Types

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

<table>
<thead>
<tr>
<th>Complex type name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetDelegateResponseMessageType</td>
<td>Specifies the response message for the GetDelegate operation.</td>
</tr>
<tr>
<td>GetDelegateType</td>
<td>Specifies the request message for the GetDelegate operation.</td>
</tr>
</tbody>
</table>

3.1.4.2.3.1 m:GetDelegateResponseMessageType Complex Type
The `GetDelegateResponseMessageType` complex type specifies the response message from the `GetDelegate` operation. The `GetDelegateResponseMessageType` complex type extends the `BaseDelegateResponseMessageType` complex type, as specified in section 2.2.4.2.

```xml
<xs:complexType name="GetDelegateResponseMessageType">
  <xs:complexContent>
    <xs:extension base="m:BaseDelegateResponseMessageType">
      <xs:sequence>
        <xs:element name="DeliverMeetingRequests" type="t:DeliverMeetingRequestsType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The following table lists the child elements of the `GetDelegateResponseMessageType` complex type.

<table>
<thead>
<tr>
<th>Element name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeliverMeetingRequests</td>
<td><code>t:DeliverMeetingRequestsType</code> (section 2.2.5.2)</td>
<td>Specifies how meeting-related messages are handled between the delegates in the response and the primary account. This element can be present. &lt;2&gt;</td>
</tr>
</tbody>
</table>

### 3.1.4.2.3.2 m:GetDelegateType Complex Type

The `GetDelegateType` complex type specifies a request to return the delegates that are users associated with a primary account’s mailbox. The `GetDelegateType` complex type extends the `BaseDelegateType` complex type, as specified in section 2.2.4.3.

```xml
<xs:complexType name="GetDelegateType">
  <xs:complexContent>
    <xs:extension base="m:BaseDelegateType">
      <xs:sequence>
        <xs:element name="UserIds" type="t:ArrayOfUserIdType" minOccurs="0"/>
        <xs:attribute name="IncludePermissions" type="xs:boolean" use="required"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The following table lists the child elements of the `GetDelegateType` complex type.
### 3.1.4.2.4 Simple Types

None.

### 3.1.4.2.5 Attributes

None.

### 3.1.4.2.6 Groups

None.

### 3.1.4.3 RemoveDelegate Operation

The **RemoveDelegate** operation removes one or more delegates from a mailbox.

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

```xml
<wsdl:operation name="RemoveDelegate">
  <wsdl:input message="tns:RemoveDelegateSoapIn" />
  <wsdl:output message="tns:RemoveDelegateSoapOut" />
</wsdl:operation>
```

The following is the WSDL binding specification of the operation.

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

### 3.1.4.3.1 Messages

The following table lists the WSDL message definitions that are specific to this operation.

<table>
<thead>
<tr>
<th>Message name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RemoveDelegateSoapIn</td>
<td>Specifies the SOAP message that removes one or more delegates from the mailbox.</td>
</tr>
<tr>
<td>RemoveDelegateSoapOut</td>
<td>Specifies the SOAP message that is returned by the server in response.</td>
</tr>
</tbody>
</table>
3.1.4.3.1.1  tns:RemoveDelegateSoapIn Message

The RemoveDelegateSoapIn WSDL message specifies the RemoveDelegate operation request to remove one or more delegates from a mailbox.

```xml
<wSDL:message name="RemoveDelegateSoapIn">
  <wSDL:part name="request" element="tns:RemoveDelegate" />
  <wSDL:part name="Impersonation" element="t:ExchangeImpersonation"/>
  <wSDL:part name="MailboxCulture" element="t:MailboxCulture"/>
  <wSDL:part name="RequestVersion" element="t:RequestServerVersion"/>
</wSDL:message>
```

The RemoveDelegateSoapIn WSDL message is the input message for the SOAP action http://schemas.microsoft.com/exchange/services/2006/messages/RemoveDelegate.

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

<table>
<thead>
<tr>
<th>Part name</th>
<th>Element/type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>request</td>
<td>tns:RemoveDelegate (section 3.1.4.3.2.2)</td>
<td>Specifies the SOAP body of the request to remove one or more delegates from a mailbox.</td>
</tr>
<tr>
<td>Impersonation</td>
<td>t:ExchangeImpersonation ([MS-OXWSCDATA] section 2.2.3.3)</td>
<td>Specifies a SOAP header that identifies the user whom the client application is impersonating.</td>
</tr>
<tr>
<td>MailboxCulture</td>
<td>t:MailboxCulture ([MS-OXWSCDATA] section 2.2.3.6)</td>
<td>Specifies a SOAP header that identifies the culture to use for accessing the mailbox. The cultures are defined by [RFC3066].</td>
</tr>
<tr>
<td>RequestVersion</td>
<td>t:RequestServerVersion ([MS-OXWSCDATA] section 2.2.3.9)</td>
<td>Specifies a SOAP header that identifies the schema version for the RemoveDelegate operation request.</td>
</tr>
</tbody>
</table>

3.1.4.3.1.2  tns:RemoveDelegateSoapOut Message

The RemoveDelegateSoapOut WSDL message specifies the server response to the RemoveDelegate operation request to remove one or more delegates from a mailbox.

```xml
<wSDL:message name="RemoveDelegateSoapOut">
  <wSDL:part name="RemoveDelegateResult" element="tns:RemoveDelegateResponse" />
  <wSDL:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wSDL:message>
```

The RemoveDelegateSoapOut WSDL message is the output message for the SOAP action http://schemas.microsoft.com/exchange/services/2006/messages/RemoveDelegate.

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

<table>
<thead>
<tr>
<th>Part name</th>
<th>Element/type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RemoveDelegateResult</td>
<td>tns:RemoveDelegateResponse (section 3.1.4.3.2.1)</td>
<td>Specifies the SOAP body of the response to a RemoveDelegate request.</td>
</tr>
<tr>
<td>ServerVersion</td>
<td>t:ServerVersionInfo ([MS-OXWSCDATA])</td>
<td>Specifies a SOAP header that</td>
</tr>
</tbody>
</table>
The **RemoveDelegate** operation returns a **RemoveDelegateResponse** element with the **ResponseClass** attribute set to "Success". The **ResponseCode** element of the **RemoveDelegateResponse** element is set to "NoError".

The operation returns the **ResponseMessages** element of the **RemoveDelegateResponse** element. A **DelegateUserResponseMessageType** element of the **ResponseMessages** element is returned for each **DelegateUser** element in the **RemoveDelegate** request. If the delegate was removed successfully, the **ResponseClass** attribute of the **DelegateUserResponseMessageType** element is set to "Success". The **ResponseCode** element of the **DelegateUserResponseMessageType** is set to "NoError".

If the delegate was not removed successfully, the **ResponseClass** attribute of the **DelegateUserResponseMessageType** element is set to "Error". The **ResponseCode** element of the **DelegateUserResponseMessageType** can be "ErrorRemoveDelegatesFailed". This error and other common errors are specified in [MS-OXWSCDATA] section 2.2.5.24.

### 3.1.4.3.2 Elements

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

<table>
<thead>
<tr>
<th>Element name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RemoveDelegate</td>
<td>Specifies a request to remove one or more delegates from a mailbox.</td>
</tr>
<tr>
<td>RemoveDelegateResponse</td>
<td>Specifies the response body content from a request to remove delegates from a mailbox.</td>
</tr>
</tbody>
</table>

#### 3.1.4.3.2.1 RemoveDelegateResponse Element

The **RemoveDelegateResponse** element specifies the response message for a **RemoveDelegate** operation.

```xml
<xs:element name="RemoveDelegateResponse"
    type="m:RemoveDelegateResponseMessageType"/>
```

#### 3.1.4.3.2.2 RemoveDelegate Element

The **RemoveDelegate** element specifies the base element for a **RemoveDelegate** operation.

```xml
<xs:element name="RemoveDelegate"
    type="m:RemoveDelegateType"/>
```

#### 3.1.4.3.3 Complex Types

The following table lists the **XML schema** complex type definitions that are specific to this operation.
<table>
<thead>
<tr>
<th>Complex type name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RemoveDelegateResponseMessageType</td>
<td>Specifies the response message for the RemoveDelegate operation.</td>
</tr>
<tr>
<td>RemoveDelegateType</td>
<td>Specifies the request message for the RemoveDelegate operation.</td>
</tr>
</tbody>
</table>

### 3.1.4.3.3.1 m:RemoveDelegateType Complex Type

The `RemoveDelegateType` complex type specifies a request to remove delegates from a primary account’s mailbox by using the `RemoveDelegate` operation. The `RemoveDelegateType` complex type extends the `BaseDelegateType` complex type, as specified in section 2.2.4.3.

```xml
<xs:complexType name="RemoveDelegateType">
  <xs:complexContent>
    <xs:extension base="m:BaseDelegateType">
      <xs:sequence>
        <xs:element name="UserIds" type="t:ArrayOfUserIdType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The following table lists the child elements of the `RemoveDelegateType` complex type.

<table>
<thead>
<tr>
<th>Element name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UserIds</td>
<td>t:ArrayOfUserIdType</td>
<td>Specifies a list of user identifiers of the delegates to remove from the mailbox.</td>
</tr>
</tbody>
</table>

### 3.1.4.3.3.2 m:RemoveDelegateResponseMessageType Complex Type

The `RemoveDelegateResponseMessageType` complex type specifies the response message from the `RemoveDelegate` operation. The `RemoveDelegateResponseMessageType` complex type extends the `BaseDelegateResponseMessageType` complex type, as specified in section 2.2.4.2.

```xml
<xs:complexType name="RemoveDelegateResponseMessageType">
  <xs:complexContent>
    <xs:extension base="m:BaseDelegateResponseMessageType"/>
  </xs:complexContent>
</xs:complexType>
```

### 3.1.4.3.4 Simple Types

None.
3.1.4.3.5 Attributes
None.

3.1.4.3.6 Groups
None.

3.1.4.4 UpdateDelegate Operation

The UpdateDelegate operation updates one or more delegates to a mailbox.

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

```xml
<wsdl:operation name="UpdateDelegate">
  <wsdl:input message="tns:UpdateDelegateSoapIn" />
  <wsdl:output message="tns:UpdateDelegateSoapOut" />
</wsdl:operation>
```

The following is the WSDL binding specification of the operation.

```xml
<wsdl:operation name="UpdateDelegate">
  <wsdl:input>
    <soap:header message="tns:UpdateDelegateSoapIn" part="Impersonation" use="literal"/>
    <soap:header message="tns:UpdateDelegateSoapIn" part="MailboxCulture" use="literal"/>
    <soap:header message="tns:UpdateDelegateSoapIn" part="RequestVersion" use="literal"/>
    <soap:body parts="request" use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap:body parts="UpdateDelegateResult" use="literal" />
    <soap:header message="tns:UpdateDelegateSoapOut" part="ServerVersion" use="literal"/>
  </wsdl:output>
</wsdl:operation>
```

3.1.4.4.1 Messages

The following table lists the WSDL message definitions that are specific to this operation.

<table>
<thead>
<tr>
<th>Message name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UpdateDelegateSoapIn</td>
<td>Specifies the SOAP message that requests that one or more delegates to a mailbox be updated.</td>
</tr>
<tr>
<td>UpdateDelegateSoapOut</td>
<td>Specifies the SOAP message that is returned by the server in response.</td>
</tr>
</tbody>
</table>

3.1.4.4.1.1 tns:UpdateDelegateSoapIn Message

The UpdateDelegateSoapIn WSDL message specifies the UpdateDelegate operation request to modify one or more delegates to a mailbox.

```xml
<wsdl:message name="UpdateDelegateSoapIn">
  <wsdl:part name="request" element="tns:UpdateDelegate" />
  <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
  <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
  <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
</wsdl:message>
```
The UpdateDelegateSoapIn WSDL message is the input message for the SOAP action http://schemas.microsoft.com/exchange/services/2006/messages/UpdateDelegate.

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

<table>
<thead>
<tr>
<th>Part name</th>
<th>Element/type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>request</td>
<td>tns:UpdateDelegate (section 3.1.4.4.2.2)</td>
<td>Specifies the SOAP body of the request to modify one or more delegates to a mailbox.</td>
</tr>
<tr>
<td>Impersonation</td>
<td>t:ExchangeImpersonation ([MS-OXWSCDATA] section 2.2.3.3)</td>
<td>Specifies a SOAP header that identifies the user whom the client application is impersonating.</td>
</tr>
<tr>
<td>MailboxCulture</td>
<td>t:MailboxCulture ([MS-OXWSCDATA] section 2.2.3.6)</td>
<td>Specifies a SOAP header that identifies the culture to use for accessing the mailbox. The cultures are defined by [RFC3066].</td>
</tr>
<tr>
<td>RequestVersion</td>
<td>t:RequestServerVersion ([MS-OXWSCDATA] section 2.2.3.9)</td>
<td>Specifies a SOAP header that identifies the schema version for the UpdateDelegate operation request.</td>
</tr>
</tbody>
</table>

3.1.4.4.1.2 tns:UpdateDelegateSoapOut message

The UpdateDelegateSoapOut WSDL message specifies the server response to the UpdateDelegate operation request to modify one or more delegates to a mailbox.

<wsdl:message name="UpdateDelegateSoapOut">
  <wsdl:part name="UpdateDelegateResult" element="tns:UpdateDelegateResponse" />
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>

The UpdateDelegateSoapOut WSDL message is the output message for the SOAP action http://schemas.microsoft.com/exchange/services/2006/messages/UpdateDelegate.

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

<table>
<thead>
<tr>
<th>Part name</th>
<th>Element/type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UpdateDelegateResult</td>
<td>tns:UpdateDelegateResponse (section 3.1.4.4.2.1)</td>
<td>Specifies the SOAP body of the response to the request to modify one or more delegates to a mailbox.</td>
</tr>
<tr>
<td>ServerVersion</td>
<td>t:ServerVersionInfo ([MS-OXWSCDATA] section 2.2.3.10)</td>
<td>Specifies a SOAP header that identifies the server version for the response.</td>
</tr>
</tbody>
</table>

The UpdateDelegate operation returns an UpdateDelegateResponse element with the ResponseClass attribute set to "Success". The ResponseCode element of the UpdateDelegateResponse element is set to "NoError".

The operation returns the ResponseMessages element of the UpdateDelegateResponse element. A DelegateUserResponseMessageType element of the ResponseMessages element is returned for each DelegateUser element in the UpdateDelegate request. If the delegate was updated successfully, the ResponseClass attribute of the DelegateUserResponseMessageType element is
set to "Success". The ResponseCode element of the DelegateUserResponseMessageType is set to "NoError".

Each successful DelegateUserResponseMessageType element includes a DelegateUser element, which includes a UserId element, a ReceiveCopiesOfMeetingRequests element, and a ViewPrivateItems element.

If the delegate was not updated successfully, the ResponseClass attribute of the DelegateUserResponseMessageType element is set to "Error". The ResponseCode element of the DelegateUserResponseMessageType can be one of the error values listed in the following table. These errors and other common errors are specified in [MS-OXWSCDATA] section 2.2.5.24.

<table>
<thead>
<tr>
<th>Error Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ErrorInvalidDelegatePermission</td>
<td>Specifies that the specified folder permissions are invalid.</td>
</tr>
<tr>
<td>ErrorUpdateDelegatesFailed</td>
<td>Specifies that the delegate list failed to be saved after delegates were updated.</td>
</tr>
</tbody>
</table>

### 3.1.4.4.2 Elements

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

<table>
<thead>
<tr>
<th>Element name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UpdateDelegate</td>
<td>Specifies a request to update one or more delegates to a mailbox.</td>
</tr>
<tr>
<td>UpdateDelegateResponse</td>
<td>Specifies the response body content from a request to update delegates to a mailbox.</td>
</tr>
</tbody>
</table>

#### 3.1.4.4.2.1 UpdateDelegateResponse Element

The UpdateDelegateResponse element specifies the response message for an UpdateDelegate operation.

```xml
<xs:element name="UpdateDelegateResponse" type="m:UpdateDelegateResponseMessageType"/>
```

#### 3.1.4.4.2.2 UpdateDelegate Element

The UpdateDelegate element specifies the base element for an UpdateDelegate operation request.

```xml
<xs:element name="UpdateDelegate" type="m:UpdateDelegateType"/>
```

#### 3.1.4.4.3 Complex Types

The following table lists the XML schema complex type definitions that are specific to this operation.
### 3.1.4.4.3.1  m:UpdateDelegateType Complex Type

The `UpdateDelegateType` complex type specifies a request to update delegates to a primary account’s mailbox by using the `UpdateDelegate` operation. The `UpdateDelegateType` complex type extends the `BaseDelegateType` complex type, as specified in section 2.2.4.3.

```xml
<xs:complexType name="UpdateDelegateType">
    <xs:complexContent>
        <xs:extension base="m:BaseDelegateType">
            <xs:sequence>
                <xs:element name="DelegateUsers" type="t:ArrayOfDelegateUserType" minOccurs="0" />
                <xs:element name="DeliverMeetingRequests" type="t:DeliverMeetingRequestsType" minOccurs="0" />
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
```

The following table lists the child elements of the `UpdateDelegateType` complex type.

<table>
<thead>
<tr>
<th>Element name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DelegateUsers</td>
<td><code>t:ArrayOfDelegateUserType</code></td>
<td>Specifies the identities of the delegates to update on a mailbox. This element can be present.</td>
</tr>
<tr>
<td>DeliverMeetingRequests</td>
<td><code>t:DeliverMeetingRequestsType</code></td>
<td>Specifies how meeting-related messages are handled between the delegate and the primary account. This element can be present.</td>
</tr>
</tbody>
</table>

### 3.1.4.4.3.2  m:UpdateDelegateResponseMessageType Complex Type

The `UpdateDelegateResponseMessageType` complex type specifies the response message from the `UpdateDelegate` operation. The `UpdateDelegateResponseMessageType` complex type extends the `BaseDelegateResponseMessageType` complex type, as specified in section 2.2.4.2.

```xml
<xs:complexType name="UpdateDelegateResponseMessageType">
    <xs:complexContent>
        <xs:extension base="m:BaseDelegateResponseMessageType" />
    </xs:complexContent>
</xs:complexType>
```
3.1.4.4 Simple Types
None.

3.1.4.4.5 Attributes
None.

3.1.4.4.6 Groups
None.

3.1.5 Timer Events
None.

3.1.6 Other Local Events
None.
4 Protocol Examples

The examples in section 4.1 through section 4.4 show the operations of this protocol.

4.1 AddDelegate Operation

The following example shows an AddDelegate request to give user1 delegate permissions on folders that are owned by user2. User1 is given Author-level permissions to user2's Calendar folder and Reviewer-level permissions to user2's Contacts folder. User1 will not receive copies of meeting messages and will be unable to view private items in user2's mailbox. Meeting requests will be sent to both user1 and user2.

```xml
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <t:RequestServerVersion Version="Exchange2013"/>
  </soap:Header>
  <soap:Body xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
    <AddDelegate>
      <Mailbox>
        <t:EmailAddress>user2@example.com</t:EmailAddress>
      </Mailbox>
      <DelegateUsers>
        <t:DelegateUser>
          <t:UserId>
            <t:PrimarySmtpAddress>user1@example.com</t:PrimarySmtpAddress>
          </t:UserId>
          <t:DelegatePermissions>
            <t:CalendarFolderPermissionLevel>Author</t:CalendarFolderPermissionLevel>
            <t:ContactsFolderPermissionLevel>Reviewer</t:ContactsFolderPermissionLevel>
          </t:DelegatePermissions>
          <t:ReceiveCopiesOfMeetingMessages>false</t:ReceiveCopiesOfMeetingMessages>
          <t:ViewPrivateItems>false</t:ViewPrivateItems>
        </t:DelegateUser>
      </DelegateUsers>
      <DeliverMeetingRequests>DelegatesAndMe</DeliverMeetingRequests>
    </AddDelegate>
  </soap:Body>
</soap:Envelope>
```

The following example shows a successful AddDelegate response to an AddDelegate request.

```xml
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Header>
    <t:ServerVersionInfo MajorVersion="8"
      MinorVersion="1"
      MajorBuildNumber="206"
      MinorBuildNumber="0"
      Version="Exchange2013"
      xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"/>
  </soap:Header>
  <soap:Body>
    <m:AddDelegateResponse
      xmlns:m="http://schemas.microsoft.com/exchange/services/2006/types"
      ResponseClass="Success"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema">
      <m:ResponseCode>NoError</m:ResponseCode>
      <m:ResponseMessages>
        <m:DelegateUserResponseMessageType ResponseClass="Success">
        </m:DelegateUserResponseMessageType>
      </m:ResponseMessages>
    </m:AddDelegateResponse>
  </soap:Body>
</soap:Envelope>
```
<m:ResponseCode>NoError</m:ResponseCode>
<m:DelegateUser>
  <t:UserId>
    <t:SID>S-1-5-21-1333220396-2200287332-232816053-1116</t:SID>
    <t:PrimarySmtpAddress>User1@example.com</t:PrimarySmtpAddress>
    <t:DisplayName>User1</t:DisplayName>
  </t:UserId>
  <t:ReceiveCopiesOfMeetingMessages>false</t:ReceiveCopiesOfMeetingMessages>
  <t:ViewPrivateItems>false</t:ViewPrivateItems>
</m:DelegateUser>
</m:DelegateUserResponseMessageType>
</m:ResponseMessages>
</m:AddDelegateResponse>
</soap:Body>
</soap:Envelope>

4.2 GetDelegate Operation

The following code example shows how to retrieve the delegate settings for all the delegates that are set on user3's mailbox. All the permissions for each user are returned in the response.

```xml
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
  <soap:Header>
    <t:RequestServerVersion Version="Exchange2013"/>
  </soap:Header>
  <soap:Body>
    <GetDelegate xmlns="http://schemas.microsoft.com/exchange/services/2006/messages"
      xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types" IncludePermissions="true">
      <Mailbox>
        <t:EmailAddress>user3@example.com</t:EmailAddress>
      </Mailbox>
    </GetDelegate>
  </soap:Body>
</soap:Envelope>
```

The following example of a GetDelegate response shows a successful response to a GetDelegate request. The response contains information about the delegate access permissions, whether the delegate can view private items, whether the delegate receives copies of meeting messages, and to whom meeting requests were delivered.

```xml
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Header>
    <t:ServerVersionInfo MajorVersion="8"
      MinorVersion="1"
      MajorBuildNumber="206"
      MinorBuildNumber="0"
      Version="Exchange2013"
      xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"/>
  </soap:Header>
  <soap:Body>
      <m:ResponseCode>NoError</m:ResponseCode>
      <m:ResponseMessages>
        <m:DelegateUserResponseMessageType ResponseClass="Success">
```

[MS-OXWSDLGM] - v20181001
Delegate Access Management Web Service Protocol
Copyright © 2018 Microsoft Corporation
Release: October 1, 2018
4.3 RemoveDelegate Operation

The following code example shows how to remove two delegates from user1's mailbox. In this example, one delegate is removed by using the delegate's primary SMTP address, and the other one is removed by using the delegate's security identifier (SID).

```xml
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/
 xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
 <soap:Header>
  <t:RequestServerVersion Version="Exchange2013"/>
 </soap:Header>
 <soap:Body>
   <Mailbox>
    <t:EmailAddress>user1@example.com</t:EmailAddress>
   </Mailbox>
   <UserIds>
    <t:UserId>
     <t:PrimarySmtpAddress>user2@example.com</t:PrimarySmtpAddress>
    </t:UserId>
    <t:UserId>
     <t:SID>S-1-5-21-1333220396-2200287332-232816053-1118</t:SID>
    </t:UserId>
   </UserIds>
  </RemoveDelegate>
 </soap:Body>
</soap:Envelope>
```

The following example of a RemoveDelegate response shows a successful response to a RemoveDelegate request. The response contains a DelegateUserResponseMessageType element for each delegate that is removed from the mailbox.

```xml
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <soap:Header>
  <t:ServerVersionInfo
   MajorVersion="8"
   MinorVersion="1">
```

[MS-OXWSDLGM] - v20181001
Delegate Access Management Web Service Protocol
Copyright © 2018 Microsoft Corporation
Release: October 1, 2018
4.4 UpdateDelegate Operation

The following example of a UpdateDelegate request shows how to update delegate permissions on user1’s account. User2 is granted the None permission level for the Tasks folder and is granted permission to view private items. User3 is granted Reviewer permissions for the Journal folder. Meeting requests are sent to the delegates, and information about the request is sent to user1.

```xml
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Header>
    <t:ServerVersionInfo MajorVersion="8"
      MinorVersion="1"
      MajorBuildNumber="206"
      MinorBuildNumber="0"
      Version="Exchange2013"
      xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"/>
  </soap:Header>
  <soap:Body>
    <m:UpdateDelegateResponse
      xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
      ResponseClass="Success"
      xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages">
      <m:ResponseCode>NoError</m:ResponseCode>
      <m:ResponseMessages>
        <m:DelegateUserResponseMessageType ResponseClass="Success">
          <m:ResponseCode>NoError</m:ResponseCode>
          <m:DelegateUser>
            <t:UserId>
              <t:SID>S-1-5-21-1333220396-2200287332-232816053-1117</t:SID>
              <t:PrimarySmtpAddress>User2@example.com</t:PrimarySmtpAddress>
              <t:DisplayName>User2</t:DisplayName>
              <t:ReceiveCopiesOfMeetingMessages>true</t:ReceiveCopiesOfMeetingMessages>
              <t:ViewPrivateItems>true</t:ViewPrivateItems>
            </t:UserId>
          </m:DelegateUser>
        </m:DelegateUserResponseMessageType>
        <m:DelegateUserResponseMessageType ResponseClass="Success">
          <m:ResponseCode>NoError</m:ResponseCode>
          <m:DelegateUser>
            <t:UserId>
              <t:SID>S-1-5-21-1333220396-2200287332-232816053-1117</t:SID>
              <t:PrimarySmtpAddress>User2@example.com</t:PrimarySmtpAddress>
              <t:DisplayName>User2</t:DisplayName>
          </t:UserId>
        </m:DelegateUserResponseMessageType>
      </m:ResponseMessages>
    </m:UpdateDelegateResponse>
  </soap:Body>
</soap:Envelope>
```
The following example shows a successful response to an UpdateDelegate operation.

```xml
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Header>
        <t:ServerVersionInfo MajorVersion="8"
            MinorVersion="1"
            MajorBuildNumber="206"
            MinorBuildNumber="0"
            Version="Exchange2013"
            xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"/>
    </soap:Header>
    <soap:Body>
        <m:UpdateDelegateResponse xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
            ResponseClass="Success" xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages">
            <m:ResponseCode>NoError</m:ResponseCode>
            <m:ResponseMessages>
                <m:DelegateUserResponseMessageType ResponseClass="Success">
                    <m:ResponseCode>NoError</m:ResponseCode>
                    <m:DelegateUser>
                        <t:UserId>
                            <t:SID>S-1-5-21-1333220396-2200287332-232816053-1117</t:SID>
                            <t:PrimarySmtpAddress>User2@example.com</t:PrimarySmtpAddress>
                            <t:DisplayName>User2</t:DisplayName>
                        </t:UserId>
                        <t:ReceiveCopiesOfMeetingMessages>true</t:ReceiveCopiesOfMeetingMessages>
                        <t:ViewPrivateItems>true</t:ViewPrivateItems>
                    </m:DelegateUser>
                </m:DelegateUserResponseMessageType>
                <m:DelegateUserResponseMessageType ResponseClass="Success">
                    <m:ResponseCode>NoError</m:ResponseCode>
                    <m:DelegateUser>
                        <t:UserId>
                            <t:SID>S-1-5-21-1333220396-2200287332-232816053-1118</t:SID>
                            <t:PrimarySmtpAddress>User3@example.com</t:PrimarySmtpAddress>
                            <t:DisplayName>User3</t:DisplayName>
                        </t:UserId>
                        <t:ReceiveCopiesOfMeetingMessages>true</t:ReceiveCopiesOfMeetingMessages>
                        <t:ViewPrivateItems>false</t:ViewPrivateItems>
                    </m:DelegateUser>
                </m:DelegateUserResponseMessageType>
            </m:ResponseMessages>
        </m:UpdateDelegateResponse>
    </soap:Body>
</soap:Envelope>
```
5 Security

5.1 Security Considerations for Implementers

None.

5.2 Index of Security Parameters

None.
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. The contents of each file are included in this section.

<table>
<thead>
<tr>
<th>File name</th>
<th>Description</th>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS-OXWSDLGM.wsdl</td>
<td>Contains the WSDL for the implementation of this protocol.</td>
<td>6</td>
</tr>
<tr>
<td>MS-OXWSDLGM-mESSAGES.xsd</td>
<td>Contains the XML schema message definitions that are used in this protocol.</td>
<td>7.1</td>
</tr>
<tr>
<td>MS-OXWSDLGM-TYPES.xsd</td>
<td>Contains the XML schema type definitions that are used in this protocol.</td>
<td>7.2</td>
</tr>
</tbody>
</table>

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-OXWSDLGM-types.xsd or MS-OXWSDLGM-mESSAGES.xsd schemas have to be placed in the common folder along with these files.

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

```xml
<?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:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages">
  <wsdl:types>
    <xs:schema id="messages" elementFormDefault="qualified" version="Exchange2016"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages">
      <xs:include schemaLocation="MS-OXWSDLGM-MESSAGES.xsd"/>
    </xs:schema>
    <xs:schema id="types" elementFormDefault="qualified" version="Exchange2016"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types">
      <xs:include schemaLocation="MS-OXWSDLGM-TYPES.xsd"/>
    </xs:schema>
  </wsdl:types>
  <wsdl:message name="GetDelegateSoapIn">
    <wsdl:part name="request" element="tns:GetDelegate"/>
  </wsdl:message>
  <wsdl:message name="GetDelegateSoapOut">
    <wsdl:part name="GetDelegateResult" element="tns:GetDelegateResponse"/>
    <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
  </wsdl:message>
  <wsdl:message name="AddDelegateSoapIn">
    <wsdl:part name="request" element="tns:AddDelegate"/>
  </wsdl:message>
  <wsdl:message name="AddDelegateSoapOut">
    <wsdl:part name="AddDelegateResult" element="tns:AddDelegateResponse"/>
    <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
  </wsdl:message>
</wsdl:definitions>
```
<wsdl:message name="RemoveDelegateSoapIn">
  <wsdl:part name="request" element="tns:RemoveDelegate" />
  <wsdl:part name="Impersonation" element="tns:ExchangeImpersonation" />
  <wsdl:part name="MailboxCulture" element="tns:MailboxCulture" />
  <wsdl:part name="RequestVersion" element="tns:RequestServerVersion" />
</wsdl:message>

<wsdl:message name="RemoveDelegateSoapOut">
  <wsdl:part name="RemoveDelegateResult" element="tns:RemoveDelegateResponse" />
</wsdl:message>

<wsdl:message name="UpdateDelegateSoapIn">
  <wsdl:part name="request" element="tns:UpdateDelegate" />
  <wsdl:part name="Impersonation" element="tns:ExchangeImpersonation" />
  <wsdl:part name="MailboxCulture" element="tns:MailboxCulture" />
  <wsdl:part name="RequestVersion" element="tns:RequestServerVersion" />
</wsdl:message>

<wsdl:message name="UpdateDelegateSoapOut">
  <wsdl:part name="UpdateDelegateResult" element="tns:UpdateDelegateResponse" />
  <wsdl:part name="ServerVersion" element="tns:ServerVersion" />
</wsdl:message>

<wsdl:portType name="ExchangeServicePortType">
  <wsdl:operation name="GetDelegate">
    <wsdl:input message="tns:GetDelegateSoapIn" />
    <wsdl:output message="tns:GetDelegateSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="AddDelegate">
    <wsdl:input message="tns:AddDelegateSoapIn" />
    <wsdl:output message="tns:AddDelegateSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="RemoveDelegate">
    <wsdl:input message="tns:RemoveDelegateSoapIn" />
    <wsdl:output message="tns:RemoveDelegateSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="UpdateDelegate">
    <wsdl:input message="tns:UpdateDelegateSoapIn" />
    <wsdl:output message="tns:UpdateDelegateSoapOut" />
  </wsdl:operation>
</wsdl:portType>
<wsdl:operation name="AddDelegate">
  <soap:operation soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/AddDelegate" />
  <wsdl:input>
    <soap:header message="tns:AddDelegateSoapIn" part="Impersonation" use="literal" /></wsdl:input>
  <wsdl:output>
    <soap:body parts="AddDelegateResult" use="literal" />
    <soap:header message="tns:AddDelegateSoapOut" part="ServerVersion" use="literal" /></wsdl:output>
</wsdl:operation>

<wsdl:operation name="RemoveDelegate">
  <wsdl:input>
    <soap:header message="tns:RemoveDelegateSoapIn" part="Impersonation" use="literal" /></wsdl:input>
  <wsdl:output>
    <soap:body parts="RemoveDelegateResult" use="literal" />
    <soap:header message="tns:RemoveDelegateSoapOut" part="ServerVersion" use="literal" /></wsdl:output>
</wsdl:operation>

<wsdl:operation name="UpdateDelegate">
  <wsdl:input>
    <soap:header message="tns:UpdateDelegateSoapIn" part="Impersonation" use="literal" /></wsdl:input>
  <wsdl:output>
    <soap:body parts="UpdateDelegateResult" use="literal" />
    <soap:header message="tns:UpdateDelegateSoapOut" part="ServerVersion" use="literal" /></wsdl:output>
</wsdl:operation>
7 Appendix B: Full XML Schema

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

<table>
<thead>
<tr>
<th>Schema name</th>
<th>Prefix</th>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Messages schema</td>
<td>m:</td>
<td>7.1</td>
</tr>
<tr>
<td>Types schema</td>
<td>t:</td>
<td>7.2</td>
</tr>
</tbody>
</table>

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

7.1 Messages Schema

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

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

<table>
<thead>
<tr>
<th>File name</th>
<th>Defining specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS-OXWSCDATA-messages.xsd</td>
<td>[MS-OXWSCDATA] section 7.1</td>
</tr>
</tbody>
</table>

```xml
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:m="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"
elementFormDefault="qualified" version="Exchange2016" id="messages">
  <xs:include schemaLocation="MS-OXWSCDATA-messages.xsd"/>
  <xs:import namespace="http://schemas.microsoft.com/exchange/services/2006/types"
schemaLocation="MS-OXWSDLGM-types.xsd"/>
  <xs:complexType name="AddDelegateResponseMessageType">
    <xs:complexContent>
      <xs:extension base="m:BaseDelegateResponseMessageType"/>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="AddDelegateResponse" type="m:AddDelegateResponseMessageType"/>
  <xs:complexType name="ArrayOfDelegateUserResponseMessageType">
    <xs:sequence>
      <xs:element name="DelegateUserResponseMessageType" type="m:DelegateUserResponseMessageType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="BaseDelegateResponseMessageType" abstract="true">
    <xs:complexContent>
      <xs:extension base="m:BaseDelegateType"/>
      <xs:sequence>
        <xs:element name="DelegateUsers" type="t:ArrayOfDelegateUserType"/>
        <xs:element name="DeliverMeetingRequests" type="t:DeliverMeetingRequestsType" minOccurs="0"/>
      </xs:sequence>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>
```
<xs:extension base="m:ResponseMessageType">
  <xs:sequence>
    <xs:element name="ResponseMessages" type="m:ArrayOfDelegateUserResponseMessageType"
      minOccurs="0"/>
  </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="BaseDelegateType" abstract="true">
  <xs:complexContent>
    <xs:extension base="m:BaseRequestType">
      <xs:sequence>
        <xs:element name="Mailbox" type="t:EmailAddressType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="DelegateUserResponseMessageType">
  <xs:complexContent>
    <xs:extension base="m:ResponseMessageType">
      <xs:sequence>
        <xs:element name="DelegateUser" type="t:DelegateUserType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="GetDelegateType">
  <xs:complexContent>
    <xs:extension base="m:BaseDelegateType">
      <xs:sequence>
        <xs:element name="UserIds" type="t:ArrayOfUserIdType" minOccurs="0"/>
        <xs:attribute name="IncludePermissions" type="xs:boolean" use="required"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:element name="GetDelegateResponse" type="m:GetDelegateResponseMessageType"/>
<xs:complexType name="GetDelegateResponseMessageType">
  <xs:complexContent>
    <xs:extension base="m:BaseDelegateResponseMessageType">
      <xs:sequence>
        <xs:element name="DeliverMeetingRequests" type="t:DeliverMeetingRequestsType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:element name="GetDelegate" type="m:GetDelegateType"/>
<xs:complexType name="RemoveDelegateType">
  <xs:complexContent>
    <xs:extension base="m:BaseDelegateType">
      <xs:sequence>
        <xs:element name="UserIds" type="t:ArrayOfUserIdType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:element name="RemoveDelegateResponse" type="m:RemoveDelegateResponseMessageType"/>
<xs:complexType name="RemoveDelegateResponseMessageType">
  <xs:complexContent>
    <xs:extension base="m:BaseDelegateResponseMessageType"/>
  </xs:complexContent>
</xs:complexType>
<xs:element name="RemoveDelegate" type="m:RemoveDelegateType"/>
<xs:complexType name="UpdateDelegateType">
  <xs:complexContent>
    <xs:extension base="m:BaseDelegateType">
      <xs:sequence>
        <xs:element name="DelegateUsers" type="t:ArrayOfDelegateUserType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:element name="UpdateDelegate" type="m:UpdateDelegateType"/>
<xs:complexType name="UpdateDelegateResponseMessageType">
  <xs:complexContent>
    <xs:extension base="m:BaseDelegateResponseMessageType"/>
  </xs:complexContent>
</xs:complexType>
<xs:element name="UpdateDelegateResponse" type="m:UpdateDelegateResponseMessageType"/>
7.2 Types Schema

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

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

<table>
<thead>
<tr>
<th>File name</th>
<th>Defining specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS-OXWSCDATA-types.xsd</td>
<td>[MS-OXWSCDATA] section 7.2</td>
</tr>
</tbody>
</table>

```xml
  <xs:include schemaLocation="MS-OXWSCDATA-types.xsd"/>
  <xs:complexType name="ArrayOfDelegateUserType">
    <xs:sequence>
      <xs:element name="DelegateUser" type="t:DelegateUserType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ArrayOfUserIdType">
    <xs:sequence>
      <xs:element name="UserId" type="t:UserIdType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="DeliverMeetingRequestsType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="DelegatesOnly"/>
      <xs:enumeration value="DelegatesAndMe"/>
      <xs:enumeration value="DelegatesAndSendInformationToMe"/>
      <xs:enumeration value="NoForward"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="DelegateUserType">
    <xs:sequence>
      <xs:element name="UserId" type="t:UserIdType"/>
      <xs:element name="DelegatePermissions" type="t:DelegatePermissionsType" minOccurs="0"/>
      <xs:element name="ReceiveCopiesOfMeetingMessages" type="xs:boolean" minOccurs="0"/>
      <xs:element name="ViewPrivateItems" type="xs:boolean" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="DelegatePermissionsType">
    <xs:sequence>
    </xs:sequence>
</xs:schema>
```
<xs:element name="CalendarFolderPermissionLevel" type="t:DelegateFolderPermissionLevelType" minOccurs="0"/>
<xs:element name="TasksFolderPermissionLevel" type="t:DelegateFolderPermissionLevelType" minOccurs="0"/>
<xs:element name="InboxFolderPermissionLevel" type="t:DelegateFolderPermissionLevelType" minOccurs="0"/>
<xs:element name="ContactsFolderPermissionLevel" type="t:DelegateFolderPermissionLevelType" minOccurs="0"/>
<xs:element name="NotesFolderPermissionLevel" type="t:DelegateFolderPermissionLevelType" minOccurs="0"/>
<xs:element name="JournalFolderPermissionLevel" type="t:DelegateFolderPermissionLevelType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="DelegateFolderPermissionLevelType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="None"/>
    <xs:enumeration value="Editor"/>
    <xs:enumeration value="Reviewer"/>
    <xs:enumeration value="Author"/>
    <xs:enumeration value="Custom"/>
  </xs:restriction>
</xs:simpleType>
</xs:schema>
Appendix C: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include updates to those products.

- Microsoft Exchange Server 2007 Service Pack 1 (SP1)
- Microsoft Exchange Server 2010
- Microsoft Exchange Server 2013
- Microsoft Exchange Server 2016
- Microsoft Exchange Server 2019

Exceptions, if any, are noted in this section. If an update version, service pack or Knowledge Base (KB) number appears with a product name, the behavior changed in that update. The new behavior also applies to subsequent updates unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

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

<1> Section 2.2.5.2: Exchange 2007 SP1 and Exchange 2010 do not use the "NoForward" enumeration value.

<2> Section 3.1.4.2.3.1: When the DeliverMeetingRequests element is set to "NoForward" for a delegate, Microsoft Exchange Server 2010 Service Pack 1 (SP1), Microsoft Exchange Server 2010 Service Pack 2 (SP2), Exchange 2013, Exchange 2016, and Exchange 2019 do not include the DeliverMeetingRequests element in the response.
9  Change Tracking

This section identifies changes that were made to this document since the last release. Changes are classified as Major, Minor, or None.

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.
- A document revision that captures changes to protocol functionality.

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 **None** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the relevant technical content is identical to the last released version.

The changes made to this document are listed in the following table. For more information, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com).

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Revision class</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>Updated supported products throughout document.</td>
<td>Major</td>
</tr>
<tr>
<td>Appendix C: Product Behavior</td>
<td>Updated list of products.</td>
<td>Major</td>
</tr>
</tbody>
</table>
10 Index

A

Abstract data model server 21
AddDelegate operation example 39
Applicability 10
Attribute groups 20
Attributes 20

C

Capability negotiation 10
Change tracking 53
Complex types 13
m:ArrayOfDelegateUserResponseMessageType Complex Type 13
m:BaseDelegateResponseMessageType Complex Type 14
m:BaseDelegateType Complex Type 14
m:DelegateUserResponseMessageType Complex Type 15
t:ArrayOfDelegateUserType Complex Type 15
t:ArrayOfUserIdType Complex Type 16
t:DelegatePermissionsType Complex Type 16
t:DelegateUserType Complex Type 17

D

Data model - abstract server 21

E

Events
local - server 38
timer - server 38
Examples
AddDelegate operation 39
GetDelegate operation 40
overview 39
RemoveDelegate operation 41
UpdateDelegate operation 42

F

Fields - vendor-extensible 11
Full WSDL 45
Full XML schema 48
Messages Schema 48
Types Schema 50

G

GetDelegate operation example 40
Glossary 7
Groups 20

I

Implementer - security considerations 44
Index of security parameters 44

Informative references 9
Initialization
server 21
Introduction 7

L

Local events server 38

M

m:ArrayOfDelegateUserResponseMessageType Complex Type complex type 13
m:BaseDelegateResponseMessageType Complex Type complex type 14
m:DelegateUserResponseMessageType Complex Type complex type 15
Message processing server 21
Messages
attribute groups 20
attributes complex types 13
elements enumerated 12
groups 20
m:ArrayOfDelegateUserResponseMessageType Complex Type complex type 13
m:BaseDelegateResponseMessageType Complex Type complex type 14
m:DelegateUserResponseMessageType Complex Type complex type 14
m:DelegateUserResponseMessageType Complex Type complex type 15
namespaces 12
simple types 18
syntax 12
t:ArrayOfDelegateUserType Complex Type complex type 15
t:ArrayOfUserIdType Complex Type complex type 16
t:DelegateFolderPermissionLevelType Simple Type simple type 18
t:DelegatePermissionsType Complex Type complex type 16
t:DelegateUserType Complex Type complex type 17
t:DeliverMeetingRequestsType Simple Type simple type 19
transport 12

N

Namespaces 12
Normative references 9

O

Operations