

# [MS-OXWSDLGM]: Delegate Access Management Web Service Protocol Specification

---

## Intellectual Property Rights Notice for Open Specifications Documentation

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

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

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

## Revision Summary

Date	Revision History	Revision Class	Comments
07/15/2009	1.0	Major	Initial Availability.
11/04/2009	1.1.0	Minor	Updated the technical content.
02/10/2010	1.2.0	Minor	Updated the technical content.
05/05/2010	1.2.1	Editorial	Revised and edited the technical content.
08/04/2010	2.0	Major	Significantly changed the technical content.
11/03/2010	3.0	Major	Significantly changed the technical content.

# Contents

<b>1 Introduction</b>	<b>6</b>
1.1 Glossary	6
1.2 References	6
1.2.1 Normative References	6
1.2.2 Informative References	7
1.3 Overview	7
1.4 Relationship to Other Protocols	7
1.5 Prerequisites/Preconditions	8
1.6 Applicability Statement	8
1.7 Versioning and Capability Negotiation	8
1.8 Vendor-Extensible Fields	8
1.9 Standards Assignments	8
<b>2 Messages</b>	<b>9</b>
2.1 Transport	9
2.2 Common Message Syntax	9
2.2.1 Namespaces	9
2.2.2 Simple Types	9
2.2.2.1 t:DelegateFolderPermissionLevelType Simple Type	9
2.2.2.2 t:DeliverMeetingRequestsType Simple Type	10
2.2.3 Complex Types	11
2.2.3.1 m:ArrayOfDelegateResponseMessageType Complex Type	12
2.2.3.2 m:BaseDelegateResponseMessageType Complex Type	12
2.2.3.3 m:BaseDelegateType Complex Type	13
2.2.3.4 m:DelegateUserResponseMessageType Complex Type	13
2.2.3.5 t:ArrayOfDelegateUserType Complex Type	14
2.2.3.6 t:ArrayOfUserIdType Complex Type	14
2.2.3.7 t:DelegatePermissionsType Complex Type	15
2.2.3.8 t:DelegateUserType Complex Type	16
2.2.4 Elements	17
2.2.5 Attributes	17
2.2.6 Groups	17
2.2.7 Attribute Groups	17
2.2.8 Messages	17
<b>3 Protocol Details</b>	<b>18</b>
3.1 ExchangeServicePortType Server Details	18
3.1.1 Abstract Data Model	18
3.1.2 Timers	18
3.1.3 Initialization	18
3.1.4 Message Processing Events and Sequencing	18
3.1.4.1 AddDelegate	18
3.1.4.1.1 Complex Types	19
3.1.4.1.1.1 m:AddDelegateResponseMessageType Complex Type	19
3.1.4.1.1.2 m:AddDelegate Complex Type	19
3.1.4.1.2 Elements	20
3.1.4.1.2.1 tns:AddDelegate Element	20
3.1.4.1.2.2 AddDelegateResponse Element	20
3.1.4.1.3 Messages	20
3.1.4.1.3.1 tns:AddDelegateSoapIn Message	20

3.1.4.1.3.2	tns:AddDelegateSoapOut Message.....	21
3.1.4.2	GetDelegate.....	21
3.1.4.2.1	Complex Types .....	21
3.1.4.2.1.1	m:GetDelegateResponseMessageType Complex Type .....	21
3.1.4.2.1.2	m:GetDelegateType Complex Type .....	22
3.1.4.2.2	Elements.....	22
3.1.4.2.2.1	tns:GetDelegate Element.....	23
3.1.4.2.2.2	GetDelegateResponse Element.....	23
3.1.4.2.3	Messages .....	23
3.1.4.2.3.1	tns:GetDelegateSoapIn Message .....	23
3.1.4.2.3.2	tns:GetDelegateSoapOut message.....	23
3.1.4.3	RemoveDelegate .....	23
3.1.4.3.1	Complex Types .....	24
3.1.4.3.1.1	m:RemoveDelegateType Complex Type.....	24
3.1.4.3.1.2	m:RemoveDelegateResponseMessageType Complex Type .....	25
3.1.4.3.2	Elements.....	25
3.1.4.3.2.1	RemoveDelegateResponse Element.....	25
3.1.4.3.2.2	RemoveDelegate Element.....	25
3.1.4.3.3	Messages .....	25
3.1.4.3.3.1	tns:RemoveDelegateSoapIn Message.....	25
3.1.4.3.3.2	tns:RemoveDelegateSoapOut message .....	26
3.1.4.4	UpdateDelegate.....	26
3.1.4.4.1	Complex Types .....	26
3.1.4.4.1.1	m:UpdateDelegateType Complex Type .....	26
3.1.4.4.1.2	m:UpdateDelegateResponseMessageType Complex Type .....	27
3.1.4.4.2	Elements.....	27
3.1.4.4.2.1	UpdateDelegateResponse Element .....	27
3.1.4.4.2.2	UpdateDelegate Element .....	28
3.1.4.4.3	Messages .....	28
3.1.4.4.3.1	tns:UpdateDelegateSoapIn Message .....	28
3.1.4.4.3.2	tns:UpdateDelegateSoapOut message.....	28
3.1.5	Timer Events .....	28
3.1.6	Other Local Events .....	28
3.2	Client Details.....	29
3.2.1	Abstract Data Model .....	29
3.2.2	Timers .....	29
3.2.3	Initialization .....	29
3.2.4	Message Processing Events and Sequencing Rules.....	29
3.2.5	Timer Events .....	29
3.2.6	Other Local Events .....	29
<b>4</b>	<b>Protocol Examples.....</b>	<b>30</b>
<b>5</b>	<b>Security.....</b>	<b>31</b>
5.1	Security Considerations for Implementers.....	31
5.2	Index of Security Parameters .....	31
<b>6</b>	<b>Appendix A: Full WSDL.....</b>	<b>32</b>
6.1	WSDL.....	32
6.2	Types Schema.....	35
6.3	Messages Schema.....	36
<b>7</b>	<b>Appendix B: Product Behavior.....</b>	<b>39</b>

<b>8 Change Tracking.....</b>	<b>40</b>
<b>9 Index .....</b>	<b>42</b>

# 1 Introduction

This document specifies the Delegate Access Management Web Service protocol, which is used to manage delegate access to **mailbox** information that is stored on a server. Clients use the SOAP protocol [[SOAP1.1](#)] to contact the delegate management service.

## 1.1 Glossary

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

**Calendar folder**  
**delegate**  
**Deleted Items folder**  
**Hypertext Transfer Protocol (HTTP)**  
**Hypertext Transfer Protocol over Secure Socket Layers (HTTPS)**  
**mailbox**  
**SOAP body**  
**SOAP fault**  
**SOAP header**  
**SOAP message**  
**Web Services Description Language (WSDL)**  
**WSDL message**  
**WSDL port type**  
**XML**  
**XML namespace**  
**XML schema**

The following terms are specific to this document:

**principal:** An entity that authorizes another entity to do work on its behalf.

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

## 1.2 References

### 1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com). We will assist you in finding the relevant information. Please check the archive site, <http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624>, as an additional source.

[MS-OXWSCDATA] Microsoft Corporation, "[Common Web Service Data Types](#)", July 2009.

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

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

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

[RFC3066] Alvestrand, H., "Tags for the Identification of Languages", BCP 47, RFC 3066, January 2001, <http://www.ietf.org/rfc/rfc3066.txt>

[SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", W3C Note, May 2000, <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

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

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

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

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

### 1.2.2 Informative References

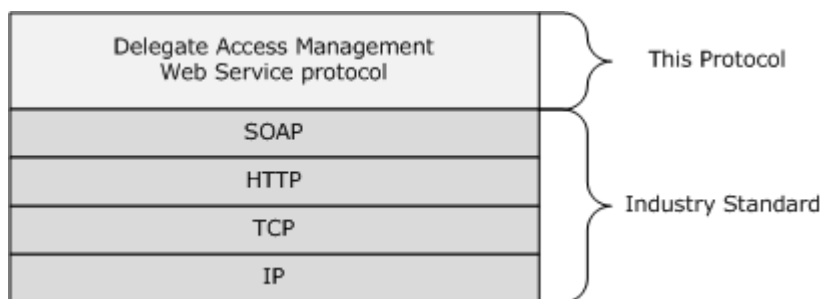
[MS-OXGLOS] Microsoft Corporation, "[Exchange Server Protocols Master Glossary](#)", April 2008.

## 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.

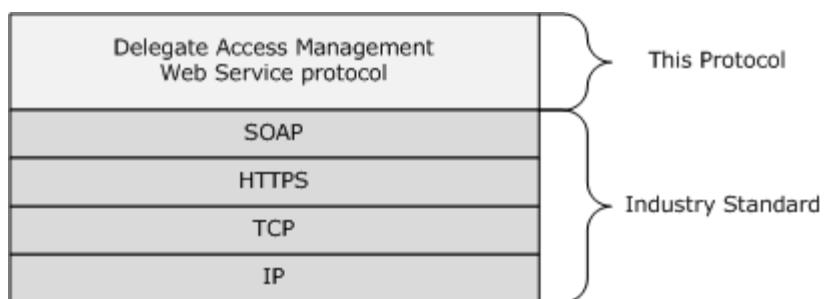
## 1.4 Relationship to Other Protocols

The Delegate Access Management Web Service protocol uses SOAP over **HTTP** [RFC2616], as shown in the following figure.



**Figure 1: Delegate Access Management Web Service protocol HTTP stack**

The Delegate Access Management Web Service protocol uses SOAP over **HTTPS** [RFC2818], as shown in the following figure.



**Figure 2: Delegate Access Management Web Service protocol HTTPS stack**

## 1.5 Prerequisites/Preconditions

None.

## 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 SOAP 1.1, as specified in section [2.1](#).
- **Protocol Versions:** The protocol specifies only one **WSDL port type** version.
- **Security and Authentication Methods:** This protocol relies on the Web server that is hosting it to perform authentication.
- **Capability Negotiation:** None.

## 1.8 Vendor-Extensible Fields

None.

## 1.9 Standards Assignments

None.



## 2 Messages

### 2.1 Transport

The SOAP version supported is SOAP 1.1. For details, see [\[SOAP1.1\]](#).

### 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.

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

#### 2.2.2 Simple Types

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

Simple Type	Description
<a href="#">t:DelegateFolderPermissionLevelType</a>	Specifies the permissions that are granted on a folder to a delegate.
<a href="#">t:DeliverMeetingRequestsType</a>	Specifies whether a delegate receives meetings on behalf of the <b>principal</b> .

##### 2.2.2.1 t:DelegateFolderPermissionLevelType Simple Type

The [DelegateFolderPermissionLevelType](#) simple type specifies the permissions that are granted on a folder to a delegate.

```
<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>

```

## Enumeration

The following values are defined by the **DelegateFolderPermissionLevelType** simple type:

Value	Description
None	Delegate has no permissions on the folder.
Editor	Delegate can read, create, and modify items in the folder.
Reviewer	Delegate can read items in the folder.
Author	Delegate can read and create items in the folder.
Custom	Delegate has a custom permission set on the folder.

### 2.2.2.2 t:DeliverMeetingRequestsType Simple Type

The [DeliverMeetingRequestsType](#) simple type specifies whether a delegate receives meetings on behalf of the principal.

```

<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>

```

## Enumeration

The following values are defined by the **DeliverMeetingRequestsType** simple type:

Value	Description
DelegatesOnly	Specifies that meeting requests are forwarded to the delegate and moved to the Deleted Items folder in the principal's mailbox.
DelegatesAndMe	Specifies that meeting requests are forwarded to the delegate and remain in the Inbox folder in the principal's mailbox.
DelegatesAndSendInformationToMe	Specifies that meeting requests are forwarded to the delegate and remain in the Inbox folder in the principal's mailbox. <a href="#">&lt;1&gt;</a>
NoForward	Specifies that meeting requests are not to be forwarded to delegates. <a href="#">&lt;2&gt;</a>

### 2.2.3 Complex Types

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

Complex Type	Description
<a href="#">m:ArrayOfDelegateUserResponseMessageType</a>	Specifies one or more delegates for mailbox.
<a href="#">m:BaseDelegateResponseMessageType</a>	Specifies the base class for all Delegate Access Management Web service protocol response types.
<a href="#">m:BaseDelegateType</a>	Specifies the base class for derived classes that specify a delegate management operation.
<a href="#">m:DelegateUserResponseMessageType</a>	Specifies a response message for a single delegate.
<a href="#">t:ArrayOfDelegateUserType</a>	Specifies one or more responses from a Delegate Access Management Web Service protocol operation.
<a href="#">t:ArrayOfUserIdType</a>	Specifies one or more user identifiers.
<a href="#">t:DelegatePermissionsType</a>	Specifies the delegate permissions for a mailbox delegate.
<a href="#">t:DelegateUserType</a>	Specifies a single delegate to add or update in a mailbox, or a delegate that is returned in a delegate management response.

### 2.2.3.1 m:ArrayOfDelegateResponseMessageType Complex Type

The [ArrayOfDelegateUserResponseMessageType](#) complex type specifies one or more responses from a Delegate Access Management Web Service protocol operation.

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

#### Child Elements

Element	Type	Description
DelegateUserResponseMessageType	<a href="#">m:DelegateUserResponseMessageType</a>	One or more responses from a Delegate Access Management Web Service protocol operation.

### 2.2.3.2 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 ([\[MS-OXWSCDATA\]](#) section 2.2.3.52).

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

#### Child Elements

Element	Type	Description
ResponseMessages	<a href="#">m:ArrayOfDelegateUserResponseMessageType</a>	One or more response messages from a Delegate Access

Element	Type	Description
		Management Web Service protocol operation. Can be present.

### 2.2.3.3 m:BaseDelegateType Complex Type

The [BaseDelegateType](#) complex type specifies the base class for derived classes that specify a delegate management operation. The BaseDelegateType complex type extends the [BaseRequestType](#) complex type ([MS-OXWSCDATA] section 2.2.3.14).

```
<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>
```

#### Child Elements

Element	Type	Description
Mailbox	<a href="#">t:EmailAddressType</a>	Specifies an e-mail address that defines the target of the delegate management operation.

### 2.2.3.4 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 ([MS-OXWSCDATA] section 2.2.3.52).

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

#### Child Elements

Element	Type	Description
DelegateUser	<a href="#">t:DelegateUserType</a>	Specifies a single delegate that is associated with a mailbox that is returned by a delegate management operation.

### 2.2.3.5 t:ArrayOfDelegateUserType Complex Type

The [ArrayOfDelegateUserType](#) complex type specifies one or more delegates for a mailbox.

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

#### Child Elements

Element	Type	Description
DelegateUser	<a href="#">t:DelegateUserType</a>	Specifies one or more mailbox delegates.

### 2.2.3.6 t:ArrayOfUserIdType Complex Type

The [ArrayOfUserIdType](#) complex type specifies one or more user identifiers.

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

#### Child Elements

Element	Type	Description
UserName	<a href="#">t:UserIdType</a>	Specifies one or more user identifiers.

### 2.2.3.7 t:DelegatePermissionsType Complex Type

The [DelegatePermissionsType](#) complex type specifies the delegate permissions for a mailbox delegate.

```
<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>
```

#### Child Elements

Element	Type	Description
CalendarFolderPermissionLevel	<a href="#">t:DelegateFolderPermissionLevelType</a>	Specifies the delegate permissions for the default <b>Calendar folder</b> . 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 <b>Inbox folder</b> . 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

Element	Type	Description
		Notes folder. Can be present.
JournalFolderPermissionLevel	t:DelegateFolderPermissionLevelType	Specifies the delegate permissions for the default Journal folder. Can be present.

### 2.2.3.8 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.

```
<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>
```

#### Child Elements

Element	Type	Description
UserId	<a href="#">t:UserIdType</a>	Specifies the user identifier for the delegate user.
DelegatePermissions	<a href="#">t:DelegatePermissionsType</a>	Specifies the delegate permissions for the delegate user. Can be present.
ReceiveCopiesOfMeetingMessages	xs:boolean	Specifies a Boolean value that indicates whether the delegate receives copies of meeting-related messages that are addressed to the principal.
ViewPrivateItems	xs:boolean	Specifies a Boolean value that indicates whether the delegate can view private calendar items on the principal's calendar.



#### **2.2.4 Elements**

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

#### **2.2.5 Attributes**

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

#### **2.2.6 Groups**

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

#### **2.2.7 Attribute Groups**

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

#### **2.2.8 Messages**

This specification does not define any common XML schema message 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.

Operation	Description
<a href="#">AddDelegate</a>	Adds a delegate to a principal's mailbox.
<a href="#">GetDelegate</a>	Gets the delegates that are associated with a principal's mailbox, if any.
<a href="#">RemoveDelegate</a>	Removes the specified delegate from the principal's mailbox.
<a href="#">UpdateDelegate</a>	Modifies the specified delegate.

#### 3.1.1 Abstract Data Model

The Delegate Access Management Web service protocol is a stateless protocol.

#### 3.1.2 Timers

None.

#### 3.1.3 Initialization

None.

#### 3.1.4 Message Processing Events and Sequencing

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

Operation	Description
<a href="#">AddDelegate</a>	Adds a delegate to a principal's mailbox.
<a href="#">GetDelegate</a>	Gets the delegates that are associated with a principal's mailbox, if any.
<a href="#">RemoveDelegate</a>	Removes the specified delegate from the principal's mailbox.
<a href="#">UpdateDelegate</a>	Modifies the specified delegate.

##### 3.1.4.1 AddDelegate

The [AddDelegate](#) operation adds a delegate to a mailbox.

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

Request

Message Format	Description
<a href="#">tns:AddDelegateSoapIn</a>	Specifies the <b>SOAP message</b> that adds the delegate to the mailbox.

Response

Message Format	Description
<a href="#">tns:AddDelegateSoapOut</a>	Specifies the SOAP message that is returned by the server in response.

### 3.1.4.1.1 Complex Types

The following XML schema complex type definitions are specific to this operation.

#### 3.1.4.1.1.1 m:AddDelegateResponseMessageType Complex Type

The [AddDelegateResponseMessageType](#) complex type specifies the response message from the [AddDelegate](#) operation (section [3.1.4.1](#)). The [AddDelegateResponseMessageType](#) complex type extends the [m:BaseDelegateResponseMessageType](#) complex type.

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

#### 3.1.4.1.1.2 m:AddDelegate Complex Type

The [AddDelegateType](#) complex type specifies a request to add delegates to a principal's **mailbox** by using the [AddDelegate](#) operation (section [3.1.4.1](#)). The [AddDelegateType](#) complex type extends the [m:BaseDelegateType](#) complex type (section [2.2.3.3](#)).

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

```

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

```

## Child Elements

Element	Type	Description
DelegateUsers	<a href="#">t:ArrayOfDelegateUserType</a>	Specifies the delegates to add to the mailbox.
DeliverMeetingRequests	<a href="#">t:DeliverMeetingRequestsType</a>	Specifies whether the delegates receive meeting-related messages that are sent to the principal.

### 3.1.4.1.2 Elements

The following XML schema element definitions are specific to this operation.

#### 3.1.4.1.2.1 tns:AddDelegate Element

The [AddDelegate](#) element specifies the base element for an [AddDelegate](#) operation (section [3.1.4.1](#)).

```

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

```

#### 3.1.4.1.2.2 AddDelegateResponse Element

The [AddDelegateResponse](#) element specifies the response message for an [AddDelegate](#) operation (section [3.1.4.1](#)).

```

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

```

### 3.1.4.1.3 Messages

The following **WSDL message** definitions are specific to this operation.

#### 3.1.4.1.3.1 tns:AddDelegateSoapIn Message

The [AddDelegateSoapIn](#) message contains four parts, as described in the following table.

Part Name	Element/Type	Description
request	<a href="#">tns:AddDelegate</a>	Specifies the request.
Impersonation	<a href="#">t:ExchangeImpersonation</a>	Specifies the user whom the client application is impersonating.

Part Name	Element/Type	Description
MailboxCulture	<a href="#">t:MailboxCulture</a>	Specifies the culture to use for accessing the mailbox. The cultures are defined by <a href="#">[RFC3066]</a> .
RequestVersion	<a href="#">t:RequestServerVersion</a>	Specifies the schema version for the <a href="#">AddDelegate</a> operation (section <a href="#">3.1.4.1</a> ) request.

### 3.1.4.1.3.2 tns:AddDelegateSoapOut Message

The [AddDelegateSoapOut](#) message contains two parts, as described in the following table.

Part Name	Element/Type	Description
AddDelegateResult	<a href="#">tns:AddDelegateResponse</a>	Specifies the response.
ServerVersion	<a href="#">t:ServerVersionInfo</a>	Specifies the server version for the response.

### 3.1.4.2 GetDelegate

The [GetDelegate](#) operation gets one or more delegates that are assigned to a mailbox.

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

Request

Message Format	Description
<a href="#">tns:GetDelegateSoapIn</a>	Specifies the SOAP message that requests the delegate.

Response

Message Format	Description
<a href="#">tns:GetDelegateSoapOut</a>	Specifies the SOAP message that is returned by the server in response.

### 3.1.4.2.1 Complex Types

The following **XML schema** complex type definitions are specific to this operation.

#### 3.1.4.2.1.1 m:GetDelegateResponseMessageType Complex Type

The **GetDelegateResponseMessageType** complex type specifies the response message from the **GetDelegate** operation. The **GetDelegateResponseMessageType** complex type extends the **m:BaseDelegateResponseMessageType** complex type (section [2.2.3.2](#)).

```
<xs:complexType name="GetDelegateResponseMessageType">
  <xs:complexContent>
    <xs:extension
```

```

    base="m:BaseDelegateResponseType"
  >
    <xs:sequence>
      <xs:element name="DeliverMeetingRequest"
        type="t:DeliverMeetingRequestsType"
        />
    </xs:sequence>
  </xs:extension>
</xs:complexContent>
</xs:complexType>

```

#### Child Elements

Element	Type	Description
DeliverMeetingRequest	<a href="#">t:DeliverMeetingRequestsType</a>	Specifies whether the delegates in the response receive meeting-related messages that are sent to the principal. This element can be present. <a href="#">&lt;3&gt;</a>

### 3.1.4.2.1.2 m:GetDelegateType Complex Type

The [GetDelegateType](#) complex type specifies a request to return the delegates that are users associated with a principal's mailbox. The GetDelegateType complex type extends the [m:BaseDelegateType](#) complex type (section [2.2.3.3](#)).

```

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

```

#### Child Elements

Element	Type	Description
UserIds	<a href="#">t:ArrayOfUserIdType</a>	Specifies one or more user identifiers for the principal users whose mailbox delegates are returned.

### 3.1.4.2.2 Elements

The following **XML schema** element definitions are specific to this operation.

### 3.1.4.2.2.1 tns:GetDelegate Element

The [GetDelegate](#) element specifies the base element for a [GetDelegate](#) operation (section [3.1.4.2](#)).

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

### 3.1.4.2.2.2 GetDelegateResponse Element

The [GetDelegateResponse](#) element specifies the response message for a [GetDelegate](#) operation (section [3.1.4.2](#)).

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

### 3.1.4.2.3 Messages

The following **WSDL** message definitions are specific to this operation.

#### 3.1.4.2.3.1 tns:GetDelegateSoapIn Message

The [GetDelegateSoapIn](#) message contains four parts, as described in the following table.

Part Name	Element/Type	Description
request	<a href="#">tns:AddDelegate</a>	Specifies the request.
Impersonation	<a href="#">t:ExchangeImpersonation</a>	Specifies the user whom the client application is impersonating.
MailboxCulture	<a href="#">t:MailboxCulture</a>	Specifies the culture to use for accessing the mailbox. The cultures are defined by <a href="#">[RFC3066]</a> .
RequestVersion	<a href="#">t:RequestServerVersion</a>	Specifies the schema version for the <a href="#">AddDelegate</a> operation (section <a href="#">3.1.4.1</a> ) request.

#### 3.1.4.2.3.2 tns:GetDelegateSoapOut message

The [GetDelegateSoapOut](#) message contains two parts, as described in the following table.

Part Name	Element/Type	Description
GetDelegateResult	<a href="#">tns:GetDelegateResponse</a>	Specifies the response.
ServerVersion	<a href="#">t:ServerVersionInfo</a>	Specifies the server version for the response.

### 3.1.4.3 RemoveDelegate

The [RemoveDelegate](#) operation removes a delegate from a mailbox.

```

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

```

#### Request

Message Format	Description
<a href="#">tns:RemoveDelegateSoapIn</a>	Specifies the SOAP message that removes the delegate from the mailbox.

#### Response

Message Format	Description
<a href="#">tns:RemoveDelegateSoapOut</a>	Specifies the SOAP message that is returned by the server in response.

### 3.1.4.3.1 Complex Types

The following **XML schema** complex type definitions are specific to this operation.

#### 3.1.4.3.1.1 m:RemoveDelegateType Complex Type

The [RemoveDelegateType](#) complex type specifies a request to remove delegates from a principal's mailbox by using the [RemoveDelegate](#) operation (section [3.1.4.3](#)). The [RemoveDelegateType](#) complex type extends the [m:BaseDelegateType](#) complex type (section [2.2.3.3](#)).

```

<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>

```

#### Child Elements

Element	Type	Description
UserIds	<a href="#">t:ArrayOfUserIdType</a>	Specifies a list of user identifiers of the delegates to remove from the mailbox.



### 3.1.4.3.1.2 m:RemoveDelegateResponseMessageType Complex Type

The [RemoveDelegateResponseMessageType](#) complex type specifies the response message from the [RemoveDelegate](#) operation (section 3.1.4.3). The RemoveDelegateResponseMessageType complex type extends the [m:BaseDelegateResponseMessageType](#) complex type (section 2.2.3.2).

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

### 3.1.4.3.2 Elements

The following **XML schema** element definitions are specific to this operation.

#### 3.1.4.3.2.1 RemoveDelegateResponse Element

The [RemoveDelegateResponse](#) element specifies the response message for a [RemoveDelegate](#) operation (section 3.1.4.3).

```
<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 (section 3.1.4.3).

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

### 3.1.4.3.3 Messages

The following **WSDL** message definitions are specific to this operation.

#### 3.1.4.3.3.1 tns:RemoveDelegateSoapIn Message

The [RemoveDelegateSoapIn](#) message contains four parts, as described in the following table.

Part Name	Element/Type	Description
request	<a href="#">tns:RemoveDelegate</a>	Specifies the request.
Impersonation		Specifies the user whom the client application is

Part Name	Element/Type	Description
	<a href="#">t:ExchangeImpersonation</a>	impersonating.
MailboxCulture	<a href="#">t:MailboxCulture</a>	Specifies the culture to use for accessing the mailbox. The cultures are defined by <a href="#">[RFC3066]</a> .
RequestVersion	<a href="#">t:RequestServerVersion</a>	Specifies the schema version for the <a href="#">AddDelegate</a> operation (section <a href="#">3.1.4.1</a> ) request.

### 3.1.4.3.3.2 tns:RemoveDelegateSoapOut message

The [RemoveDelegateSoapOut](#) message contains two parts, as described in the following table.

Part Name	Element/Type	Description
RemoveDelegateResult	<a href="#">tns:RemoveDelegateResponse</a>	Specifies the response.
ServerVersion	<a href="#">t:ServerVersionInfo</a>	Specifies the server version for the response.

### 3.1.4.4 UpdateDelegate

The [UpdateDelegate](#) operation updates a delegate to a mailbox.

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

Request

Message Format	Description
<a href="#">tns:UpdateDelegateSoapIn</a>	Specifies the SOAP message that updates the delegate to the mailbox.

Response

Message Format	Description
<a href="#">tns:UpdateDelegateSoapOut</a>	Specifies the SOAP message that is returned by the server in response.

#### 3.1.4.4.1 Complex Types

The following **XML schema** complex type definitions are specific to this operation.

##### 3.1.4.4.1.1 m:UpdateDelegateType Complex Type

The [UpdateDelegateType](#) complex type specifies a request to update delegates to a principal's mailbox by using the [UpdateDelegate](#) operation. The UpdateDelegateType complex type extends the [m:BaseDelegateType](#) complex type (section [2.2.3.3](#)).

```
<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>

```

### Child Elements

Element	Type	Description
DelegateUsers	<a href="#">t:ArrayOfDelegateUserType</a>	Specifies the identities of the delegates to update on a mailbox. This element can be present.
DeliverMeetingRequests	<a href="#">t:DeliverMeetingRequestsType</a>	Specifies a Boolean value that indicates how meeting-related messages are handled between the delegate and the principal. This element can be present.

#### 3.1.4.4.1.2 m:UpdateDelegateResponseMessageType Complex Type

The [UpdateDelegateResponseMessageType](#) complex type specifies the response message from the [UpdateDelegate](#) operation (section [3.1.4.4](#)). The [UpdateDelegateResponseMessageType](#) complex type extends the [m:BaseDelegateResponseMessageType](#) complex type (section [2.2.3.2](#)).

```

<xs:complexType name="UpdateDelegateResponseMessageType">
  <xs:complexContent>
    <xs:extension
      base="m:BaseDelegateResponseMessageType"
    />
  </xs:complexContent>
</xs:complexType>

```

#### 3.1.4.4.2 Elements

The following **XML schema** element definitions are specific to this operation.

##### 3.1.4.4.2.1 UpdateDelegateResponse Element

The [UpdateDelegateResponse](#) element specifies the response message for an [UpdateDelegate](#) operation (section [3.1.4.4](#)).

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

### 3.1.4.4.2 UpdateDelegate Element

The [UpdateDelegate](#) element specifies the base element for an [UpdateDelegate](#) operation (section [3.1.4.4](#)) request.

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

### 3.1.4.4.3 Messages

The following **WSDL** message definitions are specific to this operation.

#### 3.1.4.4.3.1 tns:UpdateDelegateSoapIn Message

The [UpdateDelegateSoapIn](#) message contains four parts, as described in the following table.

Part Name	Element/Type	Description
request	<a href="#">tns:UpdateDelegate</a>	Specifies the request.
Impersonation	<a href="#">t:ExchangeImpersonation</a>	Specifies the user whom the client application is impersonating.
MailboxCulture	<a href="#">t:MailboxCulture</a>	Specifies the culture to use for accessing the mailbox. The cultures are defined by <a href="#">[RFC3066]</a> .
RequestVersion	<a href="#">t:RequestServerVersion</a>	Specifies the schema version for the <a href="#">UpdateDelegate</a> operation (section <a href="#">3.1.4.4</a> ) request.

#### 3.1.4.4.3.2 tns:UpdateDelegateSoapOut message

The [UpdateDelegateSoapOut](#) message contains two parts, as described in the following table.

Part Name	Element/Type	Description
UpdateDelegateResult	<a href="#">tns:UpdateDelegateResponse</a>	Specifies the response.
ServerVersion	<a href="#">t:ServerVersionInfo</a>	Specifies the server version for the response.

### 3.1.5 Timer Events

None.

### 3.1.6 Other Local Events

None.

## **3.2 Client Details**

None.

### **3.2.1 Abstract Data Model**

None.

### **3.2.2 Timers**

None.

### **3.2.3 Initialization**

None.

### **3.2.4 Message Processing Events and Sequencing Rules**

None.

### **3.2.5 Timer Events**

None.

### **3.2.6 Other Local Events**

None.

## 4 Protocol Examples

None.

## **5 Security**

### **5.1 Security Considerations for Implementers**

The Delegate Access Management Web Service protocol does not use any additional security mechanisms.

### **5.2 Index of Security Parameters**

None.

## 6 Appendix A: Full WSDL

The following table lists the **XML** files that are required to implement the functionality that is specified in this document. The contents of each file are included in this section.

File name	Description	Section
MS-OXWSDLGM.wsdl	Contains the WSDL for the implementation of this protocol.	<a href="#">6.1</a>
MS-OXWSDLGM-types.xsd	Contains the XML schema type definitions that are used in this protocol.	<a href="#">6.2</a>
MS-OXWSDLGM-messages.xsd	Contains the XML schema message definitions that are used in this protocol.	<a href="#">6.3</a>

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.

### 6.1 WSDL

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

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:s="http://www.w3.org/2001/XMLSchema" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages">
  <wsdl:types>
    <xs:schema id="messages" elementFormDefault="qualified" version="Exchange2010"
xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
      <xs:include schemaLocation="MS-OXWSDLGM-messages.xsd"/>
    </xs:schema>
    <xs:schema id="types" elementFormDefault="qualified" version="Exchange2010"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
    </xs:schema>
  </wsdl:types>

  <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:portType>
</wsdl:definitions>
```



```

    <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:binding name="ExchangeServiceBinding" type="tns:ExchangeServicePortType">
    <wsdl:documentation>
      <wsi:Claim conformsTo="http://ws-i.org/profiles/basic/1.0"
xmlns:wsi="http://ws-i.org/schemas/conformanceClaim/" />
    </wsdl:documentation>
    <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http" />
    <wsdl:operation name="GetDelegate">
      <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/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>
    <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>
    <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>
<wsdl:operation name="UpdateDelegate">
    <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/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>
</wsdl:binding>

    <wsdl:message name="GetDelegateSoapIn">
        <wsdl:part name="request" element="tns:GetDelegate" />
        <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
        <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
        <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
    </wsdl:message>
    <wsdl:message name="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:part name="Impersonation" element="t:ExchangeImpersonation"/>
        <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
        <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
    </wsdl:message>
    <wsdl:message name="AddDelegateSoapOut">
        <wsdl:part name="AddDelegateResult" element="tns:AddDelegateResponse" />
        <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
    </wsdl:message>
    <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>
    <wsdl:message name="RemoveDelegateSoapOut">
        <wsdl:part name="RemoveDelegateResult" element="tns:RemoveDelegateResponse" />
        <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
    </wsdl:message>

```

```

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

```

## 6.2 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.

File name	Defining specification
MS-OXWSCDATA-types.xsd	<a href="#">[MS-OXWSCDATA]</a> section 6.3

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types"
elementFormDefault="qualified" version="Exchange2010" id="types">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <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: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>

```

### 6.3 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-types.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.

File name	Defining specification
MS-OXWSCDATA-messages.xsd	<a href="#">[MS-OXWSCDATA]</a> section 6.2

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
elementFormDefault="qualified" version="Exchange2010" 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="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>
    <xs:element name="AddDelegate" type="m:AddDelegateType"/>
    <xs:complexType name="ArrayOfDelegateUserResponseType">
        <xs:sequence>
            <xs:element name="DelegateUserResponseType"
type="m:DelegateUserResponseType" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="BaseDelegateResponseMessageType" abstract="true">
        <xs:complexContent>
            <xs:extension base="m:ResponseType">
                <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="DelegateUserResponseType">
        <xs:complexContent>
            <xs:extension base="m:ResponseType">
                <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:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>

```

```

        <xs:attribute name="IncludePermissions" type="xs:boolean"
use="required"/>
    </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="RemoveDelegate" type="m:RemoveDelegateType"/>
<xs:complexType name="RemoveDelegateResponseMessageType">
    <xs:complexContent>
        <xs:extension base="m:BaseDelegateResponseMessageType"/>
    </xs:complexContent>
</xs:complexType>
<xs:element name="RemoveDelegateResponse" type="m:RemoveDelegateResponseMessageType"/>
<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="DeliverMeetingRequests"
type="t:DeliverMeetingRequestsType" 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"/>
</xs:schema>

```

## 7 Appendix B: Product Behavior

The information in this specification is applicable to the following Microsoft products:

- Microsoft® Office Outlook® 2007
- Microsoft® Exchange Server 2007 Service Pack 1 (SP1)
- Microsoft® Outlook® 2010
- Microsoft® Exchange Server 2010
- Microsoft® Exchange Server 2010 Service Pack 1 (SP1)

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

Unless otherwise specified, any statement of optional behavior in this specification 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 product does not follow the prescription.

[<1> Section 2.2.2.2:](#) Office Outlook 2007 uses this value to remove the Accept, Tentative, and Decline buttons from the principal user's reading pane.

[<2> Section 2.2.2.2:](#) Exchange 2007 SP1 and Exchange 2010 do not use the **NoForward** enumeration value.

[<3> Section 3.1.4.2.1.1:](#) When the <DeliverMeetingRequest> element is set to "NoForward" for a delegate, Exchange 2010 and Exchange 2010 SP1 do not include the <DeliverMeetingRequest> element in the response.

## 8 Change Tracking

This section identifies changes that were made to the [MS-OXWSDLGM] protocol document between the August 2010 and November 2010 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

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

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

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

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

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

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

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

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



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

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

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

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

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

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change Type
<a href="#">Z Appendix B: Product Behavior</a>	57822 Removed Exchange 2007 from and added Exchange 2007 SP1 to the list of applicable products.	Y	Content updated.

## 9 Index

### A

[Applicability](#) 8

### C

[Capability negotiation](#) 8

[Change tracking](#) 40

Client

[abstract data model](#) 29

[initialization](#) 29

[local events](#) 29

[message processing](#) 29

[overview](#) 29

[timer events](#) 29

[timers](#) 29

### F

[Fields - vendor-extensible](#) 8

[Full WSDL](#) 32

### G

[Glossary](#) 6

### I

[Introduction](#) 6

### M

Messages

[overview](#) 9

[syntax](#) 9

[transport](#) 9

### O

[Overview \(synopsis\)](#) 7

### P

[Preconditions](#) 8

[Prerequisites](#) 8

[Product behavior](#) 39

### R

References

[informative](#) 7

[normative](#) 6

[overview](#) 6

[Relationship to other protocols](#) 7

### S

Security

[implementer considerations](#) 31

[overview](#) 31

[parameter index](#) 31

Server

[abstract data model](#) 18

[initialization](#) 18

[local events](#) 28

[message processing](#) 18

[overview](#) 18

[sequencing rules](#) 18

[timer events](#) 28

[timers](#) 18

[Standards assignments](#) 8

### T

[Tracking changes](#) 40

### V

[Vendor-extensible fields](#) 8

[Versioning](#) 8