[MS-OXWSCONT]:
Contacts Web Service Protocol

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft Open Specification Promise or the Community Promise. If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.
- **Fictitious Names.** The example companies, organizations, products, domain names, email 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	2.0.0	Major	Updated and revised the technical content.
05/05/2010	2.0.1	Editorial	Revised and edited the technical content.
08/04/2010	3.0	Major	Significantly changed the technical content.
11/03/2010	3.0	No change	No changes to the meaning, language, or formatting of the technical content.
03/18/2011	3.0	No change	No changes to the meaning, language, and formatting of the technical content.
08/05/2011	3.1	Minor	Clarified the meaning of the technical content.
10/07/2011	3.1	No change	No changes to the meaning, language, or formatting of the technical content.
01/20/2012	4.0	Major	Significantly changed the technical content.
04/27/2012	4.0	No change	No changes to the meaning, language, or formatting of the technical content.
07/16/2012	4.1	Minor	Clarified the meaning of the technical content.
10/08/2012	4.2	Minor	Clarified the meaning of the technical content.
02/11/2013	5.0	Major	Significantly changed the technical content.
07/26/2013	6.0	Major	Significantly changed the technical content.
11/18/2013	7.0	Major	Significantly changed the technical content.
02/10/2014	7.0	No change	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1 Introduction

This document specifies the Contacts Web Service protocol.

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

1.1 Glossary

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

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

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

base64 encoding
contact
distribution list
endpoint
mailbox
message store
permission
S/MIME (Secure/Multipurpose Internet Mail Extensions)
Simple Mail Transfer Protocol (SMTP)
Uniform Resource Identifier (URI)
Uniform Resource Locator (URL)
web server
Web Services Description Language (WSDL)
WSDL message
WSDL port type
XML namespace prefix
XML schema

The following terms are specific to this document:

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

1.2 References

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

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information.

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

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

[MS-OXWSDLIST] Microsoft Corporation, "Distribution List Creation and Usage Web Service Protocol".

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

[MS-OXWSRSLNM] Microsoft Corporation, "Resolve Recipient Names Web Service Protocol".

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

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

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

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

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

[WSIBASIC] Ballinger, K., Ehnebuske, D., Gudgin, M., et al., Eds., "Basic Profile Version 1.0", Final Material, April 2004, http://www.ws-i.org/Profiles/BasicProfile-1.0-2004-04-16.html

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

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

[XMLSCHEMA2] Biron, P.V., 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-GLOS] Microsoft Corporation, "Windows Protocols Master Glossary".

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

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

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

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

1.3 Overview

The Contacts Web Service protocol provides the messages needed to create, get, update, delete, move, and copy **contact** items on the server.

1.4 Relationship to Other Protocols

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

This protocol uses **SOAP**, as described in [SOAP1.1], to specify the structure information that is exchanged between the client and the server. This protocol uses the **XML schema**, as described in [XMLSCHEMA1] and [XMLSCHEMA2], to describe the message content that is sent to and from the server.

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

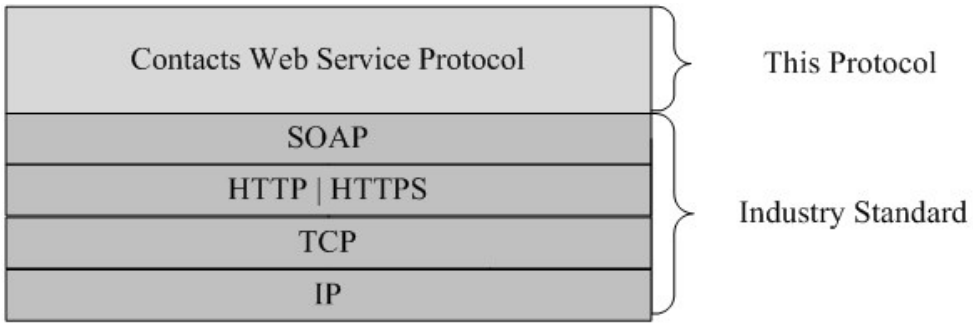

Figure 1: This protocol in relation to other protocols

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

1.5 Prerequisites/Preconditions

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

1.6 Applicability Statement

This protocol is applicable to client programs that create, update, or manage contact items in the server **message store**.

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

- **Supported Transports:** This protocol uses SOAP 1.1. For more information, see section 2.1.

- **Protocol Versions:** This 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.
- **Localization:** This protocol uses the **MailboxCulture** part, as described in [MS-OXWSCORE] section 3.1.4.1.1.1, to specify the culture of a **mailbox**, and elements that are of the **xs:dateTime** type, as described in section 2.2.4.2.
- **Capability Negotiation:** None.

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, as specified in [SOAP1.1].

The protocol MUST support SOAP over HTTP, as specified in [RFC2616]. The protocol SHOULD use secure communications by means of HTTPS, as specified in [RFC2818].

2.2 Common Message Syntax

This section contains common definitions that are used by this protocol. The syntax of the definitions uses XML schema, as 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	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
tns	http://schemas.microsoft.com/exchange/services/2006/messages	
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA2]
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
t	http://schemas.microsoft.com/exchange/services/2006/types	
wsi	http://ws-i.org/schemas/conformanceClaim/	[WSIBASIC]

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.

Complex type	Description
t:ArrayOfBinaryType (section 2.2.4.1)	Specifies a collection of certificates for a contact.
t:ContactItemType (section 2.2.4.2)	Represents a server contact item.

2.2.4.1 t:ArrayOfBinaryType Complex Type

The **ArrayOfBinaryType** complex type specifies a collection of certificates for a contact.<1> This type is used by the **UserSMIMECertificate** element and the **MSExchangeCertificate** element of the **ContactItemType** complex type, as specified in section 2.2.4.2.

```
<xs:complexType name="ArrayOfBinaryType">
  <xs:sequence>
    <xs:element name="Base64Binary" type="xs:base64Binary" minOccurs="0"
maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>
```

Child element

Element name	Type	Description
Base64Binary	**xs:base64Binary** [XMLSCHEMA2]	Specifies a single certificate for a contact. The value is encoded with **base64 encoding**.

2.2.4.2 t:ContactItemType Complex Type

The **ContactItemType** complex type represents a server contact item. It is also used by the **ResolveNames** method ([MS-OXWSRSLNM] section 3.1.4.1), returning directory and store contacts matching a search string. This type extends the **ItemType** complex type, as specified in [MS-OXWSCORE] section 2.2.4.22. This type is used by the **CreateItem** operation, as specified in section 3.1.4.6, and the **UpdateItem** operation, as specified in section 3.1.4.3.

```
<xs:complexType name="t:ContactItemType">
  <xs:complexContent>
    <xs:extension
      base="t:ItemType"
    >
      <xs:sequence>
        <xs:element name="FileAs"
          type="xs:string"
          minOccurs="0"
         />
        <xs:element name="FileAsMapping"
          type="t:FileAsMappingType"
         />
        <xs:element name="DisplayName"
```

```
  type="xs:string"
 />
<xs:element name="GivenName"
  type="xs:string"
 />
<xs:element name="Initials"
  type="xs:string"
 />
<xs:element name="MiddleName"
  type="xs:string"
 />
<xs:element name="Nickname"
  type="xs:string"
 />
<xs:element name="CompleteName"
  type="t:CompleteNameType"
 />
<xs:element name="CompanyName"
  type="xs:string"
 />
<xs:element name="EmailAddresses"
  type="t:EmailAddressDictionaryType"
 />
<xs:element name="PhysicalAddresses"
  type="t:PhysicalAddressDictionaryType"
 />
<xs:element name="PhoneNumbers"
  type="t:PhoneNumberDictionaryType"
 />
<xs:element name="AssistantName"
  type="xs:string"
 />
<xs:element name="Birthday"
  type="xs:dateTime"
 />
<xs:element name="BusinessHomePage"
  type="xs:anyURI"
 />
<xs:element name="Children"
  type="t:ArrayOfStringsType"
 />
<xs:element name="Companies"
  type="t:ArrayOfStringsType"
 />
<xs:element name="ContactSource"
  type="t:ContactSourceType"
 />
<xs:element name="Department"
  type="xs:string"
 />
<xs:element name="Generation"
  type="xs:string"
 />
<xs:element name="ImAddresses"
  type="t:ImAddressDictionaryType"
 />
<xs:element name="JobTitle"
  type="xs:string"
 />
```

```
<xs:element name="Manager"
  type="xs:string"
 />
<xs:element name="Mileage"
  type="xs:string"
 />
<xs:element name="OfficeLocation"
  type="xs:string"
 />
<xs:element name="PostalAddressIndex"
  type="t:PhysicalAddressIndexType"
 />
<xs:element name="Profession"
  type="xs:string"
 />
<xs:element name="SpouseName"
  type="xs:string"
 />
<xs:element name="Surname"
  type="xs:string"
 />
<xs:element name="WeddingAnniversary"
  type="xs:dateTime"
 />
<xs:element name="HasPicture"
  type="xs:boolean"
 />
<xs:element name="PhoneticFullName"
  type="xs:string"
  minOccurs="0"
/>
<xs:element name="PhoneticFirstName"
  type="xs:string"
  minOccurs="0"
/>
<xs:element name="PhoneticLastName"
  type="xs:string"
  minOccurs="0"
/>
<xs:element name="Alias"
  type="xs:string"
  minOccurs="0"
/>
<xs:element name="Notes"
  type="xs:string"
  minOccurs="0"
/>
<xs:element name="Photo"
  type="xs:base64Binary"
  minOccurs="0"
/>
<xs:element name="UserSMIMECertificate"
  type="t:ArrayOfBinaryType"
  minOccurs="0"
/>
<xs:element name="MSExchangeCertificate"
  type="t:ArrayOfBinaryType"
  minOccurs="0"
/>
```

```
          <xs:element name="DirectoryId"
            type="xs:string"
            minOccurs="0"
          />
          <xs:element name="ManagerMailbox"
            type="t:SingleRecipientType"
            minOccurs="0"
          />
          <xs:element name="DirectReports"
            type="t:ArrayOfRecipientsType"
            minOccurs="0"
          />
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
```

Child Elements<2>

Element name	Type	Description
FileAs	**xs:string** [XMLSCHEMA2]	Represents how a contact is filed in the Contacts folder.
FileAsMapping	**t:FileAsMappingType** (section 3.1.4.1.2.2)	Defines how to construct what is displayed for a contact.
DisplayName	**xs:string**	Contains the display name of a contact.
GivenName	**xs:string**	Contains the given name for a contact.
Initials	**xs:string**	Contains the initials for a contact.
MiddleName	**xs:string**	Represents the middle name of a contact.
Nickname	**xs:string**	Represents the nickname of a contact.
CompleteName	**t:CompleteNameType** (section 3.1.4.1.1.1)	Represents the complete name of a contact.
CompanyName	**xs:string**	Contains the company name that is associated with a contact.
EmailAddresses	**t:EmailAddressDictionaryType** (section 3.1.4.1.1.5)	Contains e-mail addresses that are associated with a contact.
PhysicalAddresses	**t:PhysicalAddressDictionaryType** (section 3.1.4.1.1.11)	Represents a collection of physical addresses that are associated with a contact.
PhoneNumbers	**t:PhoneNumberDictionaryType** (section 3.1.4.1.1.9)	Represents a collection of telephone numbers for a contact.
AssistantName	**xs:string**	Contains the name of the assistant for the contact.

Element name	Type	Description
Birthday	**xs:dateTime** [XMLSCHEMA2]	Represents the birthday of the contact.
BusinessHomePage	**xs:anyURI** [XMLSCHEMA2]	Contains the business home page **Uniform Resource Identifier (URI)** of a contact.
Children	**t:ArrayOfStringsType** ([MS-OXWSCDATA] section 2.2.4.11)	Contains the names of children for the contact.
Companies	**t:ArrayOfStringsType**	Contains the names of companies that are associated with a contact.
ContactSource	**t:ContactSourceType** (section 2.2.5.1)	Describes whether the contact is located in the server message store or the directory service.
Department	**xs:string**	Contains the work department for the contact.
Generation	**xs:string**	Contains a generational abbreviation that follows the full name of a contact.
ImAddresses	**t:ImAddressDictionaryType** (section 3.1.4.1.1.7)	Contains instant messaging addresses for a contact.
JobTitle	**xs:string**	Contains the job title of a contact.
Manager	**xs:string**	Represents the manager of a contact.
Mileage	**xs:string**	Represents the mileage for a contact.
OfficeLocation	**xs:string**	Represents the office location of a contact.
PostalAddressIndex	**t:PhysicalAddressIndexType** (section 3.1.4.1.2.5)	Represents the index of one of the physical addresses, which is a contact's mailing address.
Profession	**xs:string**	Represents the profession of a contact.
SpouseName	**xs:string**	Represents the name of the spouse/partner of a contact.
Surname	**xs:string**	Contains the surname of a contact.
WeddingAnniversary	**xs:dateTime**	Contains the wedding anniversary date of a contact.
HasPicture	**xs:boolean** [XMLSCHEMA2]	Represents that the contact has a picture.
PhoneticFullName	**xs:string**	Contains the full name of a contact, including the first and

Element name	Type	Description
		last name, spelled phonetically.
PhoneticFirstName	**xs:string**	Contains the first name of a contact, spelled phonetically.
PhoneticLastName	**xs:string**	Contains the last name of a contact, spelled phonetically.
Alias	**xs:string**	Contains the email alias of a contact.
Notes	**xs:string**	Contains supplementary contact information.
Photo	**xs:base64Binary** [XMLSCHEMA2]	Contains a value that encodes the photo of a contact.
UserSMIMECertificate	**t:ArrayOfBinaryType** (section 2.2.4.1)	Contains a value that encodes a contacts **S/MIME** certificate.
MSExchangeCertificate	**t:ArrayOfBinaryType**	Contains a value that encodes the server certificate of a contact.
DirectoryId	**xs:string**	Contains the directory identifier of a contact.
ManagerMailbox	**t:SingleRecipientType** ([MS-OXWSCDATA] section 2.2.4.60)	Contains **SMTP** information that identifies the mailbox of a contact's manager.
DirectReports	**t:ArrayOfRecipientsType** ([MS-OXWSCDATA] section 2.2.4.9)	Contains SMTP information that identifies the mailboxes of a contact's direct reports.

2.2.5 Simple Types

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

Simple Type	Description
t:ContactSourceType (section 2.2.5.1)	Specifies whether a contact or **distribution list** is located in the server database or in the directory service.

2.2.5.1 t:ContactSourceType Simple Type

The **ContactSourceType** specifies whether a contact or distribution list is located in the server database or in the directory service.

```
<xs:simpleType name="ContactSourceType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="ActiveDirectory"
```

```
      />
     <xs:enumeration
       value="Store"
      />
   </xs:restriction>
 </xs:simpleType>
```

Enumeration

The following values are defined by the **ContactSourceType** simple type.

Value	Meaning
ActiveDirectory	Specifies that the contact or distribution list is located in the directory service.
Store	Specifies that the contact or distribution list is located in the server database.

This is applicable to the Contacts Web Service protocol and to the Distribution List Creation and Usage Web Service protocol [MS-OXWSDLIST].

It is also used by the **ResolveNames** method ([MS-OXWSRSLNM] section 3.1.4.1), returning directory and store contacts matching a search string.

2.2.6 Attributes

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

2.2.7 Groups

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

2.2.8 Attribute Groups

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

3 Protocol Details

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

3.1 ExchangeServicePortType Server Details

This protocol defines a single port type.

3.1.1 Abstract Data Model

None.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing Rules

This protocol includes the operations listed in the following table.

Operation	Description
CopyItem (section 3.1.4.5)	Defines a request to copy an item in a mailbox in the server.
CreateItem (section 3.1.4.6)	Defines a request to create an item in the server.
DeleteItem (section 3.1.4.2)	Defines a request to delete an item from a mailbox in the server.
GetItem (section 3.1.4.1)	Defines a request to get an item from a mailbox in the server.
MoveItem (section 3.1.4.4)	Defines a request to move an item in the server.
UpdateItem (section 3.1.4.3)	Defines a request to update an item in a mailbox.

3.1.4.1 GetItem

This protocol uses the **GetItem** operation specified in [MS-OXWSCORE] section 3.1.4.4 to get contact item elements.

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

Request

Message Format	Description
tns:GetItemSoapIn ([MS-OXWSCORE] section 3.1.4.4.1.1)	Specifies the **SOAP message** that defines the contact item to get. The **Items** ([MS-OXWSCDATA] section 2.2.4.42) child element of the **GetItem** ([MS-OXWSCORE] section 3.1.4.4.2.1) child element that specifies the **XML** request MUST contain the following elements: **t:ItemResponseShapeType** ([MS-OXWSCDATA] section 2.2.4.38), **t:ItemIdType** ([MS-OXWSCORE] section 2.2.4.23). All other elements MUST be empty.

Response

Message Format	Description
tns:GetItemSoapOut ([MS-OXWSCORE] section 3.1.4.4.1.2)	Specifies the SOAP message 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 t:CompleteNameType Complex Type

The **CompleteNameType** complex type represents the complete name of a contact.

```
<xs:complexType name="CompleteNameType">
  <xs:sequence>
    <xs:element name="Title"
      type="xs:string"
      minOccurs="0"
     />
    <xs:element name="FirstName"
      type="xs:string"
      minOccurs="0"
     />
    <xs:element name="MiddleName"
      type="xs:string"
      minOccurs="0"
     />
    <xs:element name="LastName"
      type="xs:string"
      minOccurs="0"
     />
    <xs:element name="Suffix"
      type="xs:string"
      minOccurs="0"
     />
    <xs:element name="Initials"
      type="xs:string"
      minOccurs="0"
     />
    <xs:element name="FullName"
      type="xs:string"
      minOccurs="0"
     />
    <xs:element name="Nickname"
      type="xs:string"
      minOccurs="0"
```

```
      />
     <xs:element name="YomiFirstName"
       type="xs:string"
       minOccurs="0"
      />
     <xs:element name="YomiLastName"
       type="xs:string"
       minOccurs="0"
      />
   </xs:sequence>
 </xs:complexType>
```

Child Elements

Element name	Type	Description
Title	**xs:string** [XMLSCHEMA2]	Contains the title of a contact.
FirstName	**xs:string**	Contains the first name of a contact. This is the same as GivenName.
MiddleName	**xs:string**	Contains the middle name of a contact.
LastName	**xs:string**	Contains the last name of a contact. This is the same as the Surname.
Suffix	**xs:string**	Contains a suffix to a contact's name. This is the same as the Generation property.
Initials	**xs:string**	Contains the initials of a contact.
FullName	**xs:string**	Contains the full name of a contact.
Nickname	**xs:string**	Contains the nickname of a contact.
YomiFirstName	**xs:string**	Contains the name used in Japan for the searchable or phonetic spelling of a Japanese first name.
YomiLastName	**xs:string**	Contains the name used in Japan for the searchable or phonetic spelling of a Japanese last name.

3.1.4.1.1.2 t:ContactsFolderType Complex Type

The **ContactsFolderType** complex type represents a Contacts folder in a mailbox.

```
<xs:complexType name="ContactsFolderType">
  <xs:complexContent>
    <xs:extension
      base="t:BaseFolderType"
    >
      <xs:sequence>
        <xs:element name="SharingEffectiveRights"
          type="t:PermissionReadAccessType"
          minOccurs="0"
```

```
          />
          <xs:element name="PermissionSet"
            type="t:PermissionSetType"
            minOccurs="0"
          />
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
```

Child Elements

Element name	Type	Description
SharingEffectiveRights	**t:PermissionReadAccessType** ([MS-OXWSFOLD] section 2.2.5.4)	Specifies whether a user has **permission** to read items in a folder.
PermissionSet	**t:PermissionSetType** ([MS-OXWSFOLD] section 2.2.4.12)	Contains all the permissions that are configured for a folder.

3.1.4.1.1.3 t:ContactsViewType Complex Type

The **ContactsViewType** complex type represents the settings that are used to return contact items based on their alphabetical display names.

```
<xs:complexType name="ContactsViewType">
  <xs:complexContent>
    <xs:extension
      base="t:BasePagingType"
    >
      <xs:attribute name="InitialName"
        type="xs:string"
        use="optional"
       />
      <xs:attribute name="FinalName"
        type="xs:string"
        use="optional"
       />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Attributes

Attribute name	Type	Description
InitialName	**xs:string** [XMLSCHEMA2]	Contains the first name in a contacts list to return in a response.
FinalName	**xs:string**	Contains the last name in a contacts list to return in a response.

3.1.4.1.1.4 t:EmailAddressDictionaryEntryType Complex Type

The **EmailAddressDictionaryEntryType** complex type represents an e-mail address that is associated with a contact.

```
<xs:complexType name="EmailAddressDictionaryEntryType">
  <xs:simpleContent>
    <xs:extension
      base="xs:string"
    >
      <xs:attribute name="Key"
        type="t:EmailAddressKeyType"
        use="required"
       />
      <xs:attribute name="Name"
        type="xs:string"
        use="optional"
       />
      <xs:attribute name="RoutingType"
        type="xs:string"
        use="optional"
       />
      <xs:attribute name="MailboxType"
        type="t:MailboxTypeType"
        use="optional"
       />
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Attributes

Name	Type	Description
Key	**t:EmailAddressKeyType** (section 3.1.4.1.2.1)	Contains a value that identifies an e-mail address that is associated with a contact.
Name<3>	**xs:string** [XMLSCHEMA2]	Contains the display name associated with an e-mail address of the contact.
RoutingType<4>	**xs:string**	Contains the routing type associated with an e-mail address of the contact.
MailboxType<5>	**t:MailboxTypeType** ([MS-OXWSCDATA] section 2.2.5.18)	Contains the type of mailbox that is represented by the e-mail address of the contact.

3.1.4.1.1.5 t:EmailAddressDictionaryType Complex Type

The **EmailAddressDictionaryType** complex type contains e-mail addresses.

```
<xs:complexType name="EmailAddressDictionaryType">
  <xs:sequence>
    <xs:element name="Entry"
      type="t:EmailAddressDictionaryEntryType"
      maxOccurs="unbounded"
```

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

Child Elements

Element name	Type	Description
Entry	**t:EmailAddressDictionaryEntryType** (section 3.1.4.1.1.4)	Represents an e-mail address that is associated with a contact.

3.1.4.1.1.6 t:ImAddressDictionaryEntryType Complex Type

The **ImAddressDictionaryEntryType** complex type represents a collection of instant messaging addresses for a contact.

```
<xs:complexType name="ImAddressDictionaryEntryType">
  <xs:simpleContent>
    <xs:extension
      base="xs:string"
    >
      <xs:attribute name="key"
        type="t:ImAddressKeyType"
        use="required"
       />
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Attributes

Attribute name	Type	Description
key	**t:ImAddressKeyType** (section 3.1.4.1.2.3)	Represents the instant messaging addresses for a contact.

3.1.4.1.1.7 t:ImAddressDictionaryType Complex Type

The **ImAddressDictionaryType** complex type contains instant messaging addresses for a contact.

```
<xs:complexType name="ImAddressDictionaryType">
  <xs:sequence>
    <xs:element name="Entry"
      type="t:ImAddressDictionaryEntryType"
      maxOccurs="unbounded"
     />
  </xs:sequence>
</xs:complexType>
```

Child Elements

Element name	Type	Description
Entry	**t:ImAddressDictionaryEntryType** (section 3.1.4.1.1.8)	Represents a collection of instant messaging addresses for a contact.

3.1.4.1.1.8 t:PhoneNumberDictionaryEntryType Complex Type

The **PhoneNumberDictionaryEntryType** complex type contains a telephone number for a contact.

```
<xs:complexType name="PhoneNumberDictionaryEntryType">
  <xs:simpleContent>
    <xs:extension
      base="xs:string"
    >
      <xs:attribute name="Key"
        type="t:PhoneNumberKeyType"
        use="required"
       />
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Attributes

Attribute name	Type	Description
Key	**t:PhoneNumberKeyType** (section 3.1.4.1.2.4)	Represents types of telephone numbers for a contact.

3.1.4.1.1.9 t:PhoneNumberDictionaryType Complex Type

The **PhoneNumberDictionaryType** complex type represents telephone numbers for a contact.

```
<xs:complexType name="PhoneNumberDictionaryType">
  <xs:sequence>
    <xs:element name="Entry"
      type="t:PhoneNumberDictionaryEntryType"
      maxOccurs="unbounded"
     />
  </xs:sequence>
</xs:complexType>
```

Child Elements

Element name	Type	Description
Entry	**t:PhoneNumberDictionaryEntryType** (section	Contains a telephone number for

Element name	Type	Description
	3.1.4.1.1.8)	a contact.

3.1.4.1.1.10 t:PhysicalAddressDictionaryEntryType Complex Type

The **PhysicalAddressDictionaryEntryType** complex type contains information that defines a physical address, such as a street address.

```
<xs:complexType name="PhysicalAddressDictionaryEntryType">
  <xs:sequence>
    <xs:element name="Street"
      type="xs:string"
      minOccurs="0"
     />
    <xs:element name="City"
      type="xs:string"
      minOccurs="0"
     />
    <xs:element name="State"
      type="xs:string"
      minOccurs="0"
     />
    <xs:element name="CountryOrRegion"
      type="xs:string"
      minOccurs="0"
     />
    <xs:element name="PostalCode"
      type="xs:string"
      minOccurs="0"
     />
  </xs:sequence>
  <xs:attribute name="Key"
    type="t:PhysicalAddressKeyType"
    use="required"
   />
</xs:complexType>
```

Child Elements

Element name	Type	Description
Street	**xs:string** [XMLSCHEMA2]	Contains the street address for a contact item.
City	**xs:string**	Contains the city name for a contact item.
State	**xs:string**	Contains the state for a contact item.
CountryOrRegion	**xs:string**	Contains the country or region for a contact item.
PostalCode	**xs:string**	Contains the postal code for a contact item.

Attributes

Attribute name	Type	Description
Key	**t:PhysicalAddressKeyType** (section 3.1.4.1.2.6)	Identifies the types of physical addresses for a contact.

3.1.4.1.1.11 t:PhysicalAddressDictionaryType Complex Type

The **PhysicalAddressDictionaryType** complex type contains physical addresses that are associated with a contact.

```
<xs:complexType name="PhysicalAddressDictionaryType">
  <xs:sequence>
    <xs:element name="Entry"
      type="t:PhysicalAddressDictionaryEntryType"
      maxOccurs="unbounded"
     />
  </xs:sequence>
</xs:complexType>
```

Child Elements

Element name	Type	Description
Entry	**t:PhysicalAddressDictionaryEntryType** (section 3.1.4.1.1.10)	Contains information that defines a physical address, such as a street address.

3.1.4.1.2 Simple Types

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

3.1.4.1.2.1 t:EmailAddressKeyType Simple Type

The **EmailAddressKeyType** simple type represents a way to identify a single e-mail address within the e-mail address collection for a contact.

```
<xs:simpleType name="EmailAddressKeyType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="EmailAddress1"
     />
    <xs:enumeration
      value="EmailAddress2"
     />
    <xs:enumeration
      value="EmailAddress3"
     />
  </xs:restriction>
</xs:simpleType>
```

Enumeration

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

Value	Meaning
EmailAddress1	Identifies the first e-mail address for the contact.
EmailAddress2	Identifies the second e-mail address for the contact.
EmailAddress3	Identifies the third e-mail address for the contact.

3.1.4.1.2.2 t:FileAsMappingType Simple Type

The **FileAsMappingType** simple type defines how to construct what is displayed for a contact in the **FileAs** property.

```
<xs:simpleType name="FileAsMappingType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="None"
     />
    <xs:enumeration
      value="LastCommaFirst"
     />
    <xs:enumeration
      value="FirstSpaceLast"
     />
    <xs:enumeration
      value="Company"
     />
    <xs:enumeration
      value="LastCommaFirstCompany"
     />
    <xs:enumeration
      value="CompanyLastFirst"
     />
    <xs:enumeration
      value="LastFirst"
     />
    <xs:enumeration
      value="LastFirstCompany"
     />
    <xs:enumeration
      value="CompanyLastCommaFirst"
     />
    <xs:enumeration
      value="LastFirstSuffix"
     />
    <xs:enumeration
      value="LastSpaceFirstCompany"
     />
    <xs:enumeration
```

```
        value="CompanyLastSpaceFirst"
       />
      <xs:enumeration
        value="LastSpaceFirst"
       />
      <xs:enumeration
        value="DisplayName"
       />
      <xs:enumeration
        value="FirstName"
       />
      <xs:enumeration
        value="LastFirstMiddleSuffix"
       />
      <xs:enumeration
        value="LastName"
       />
      <xs:enumeration
        value="Empty"
       />
    </xs:restriction>
  </xs:simpleType>
```

Enumeration

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

Value	Meaning
None	Indicates that the **FileAs** value is not constructed from properties of other contacts, but is represented by a string, saved "as is".
LastCommaFirst	Indicates that the contact is displayed as the last name followed by a comma, a space, the first name, a space, and the middle name.
FirstSpaceLast	Indicates that the contact is displayed as the first name followed by a space, the middle name, a space, the last name, a space, and the suffix for the contact.
Company	Indicates that the company name is displayed.
LastCommaFirstCompany	Indicates that the contact is displayed as the last name, a comma, a space, the first name, a space, the middle name, a space, a left parenthesis, the company name, and a right parenthesis.
CompanyLastFirst	Indicates that the contact is displayed as the company name, a space, a left parenthesis, the last name, the first name, a space, the middle name, and a right parenthesis.
LastFirst	Indicates that the contact is displayed as the last name followed by the first name, a space, and the middle name.
LastFirstCompany	Indicates that the contact is displayed as the last name, the first name, a space, the middle name, a space, a left parenthesis, the company name, and a right parenthesis.
CompanyLastCommaFirst	Indicates that the contact is displayed as the company name, a space, a left parenthesis, the last name, a comma, a space, and the first name, a space,

Value	Meaning
	the middle name, and a right parenthesis.
LastFirstSuffix	Indicates that the contact is displayed as the last name, the first name, a space, and the suffix for the contact.
LastSpaceFirstCompany	Indicates that the contact is displayed as the last name, a space, the first name, a space, the middle name, a space, a left parenthesis, the company name, and a right parenthesis.
CompanyLastSpaceFirst	Indicates that the contact is displayed as the company name, a space, a left parenthesis, the last name, a space, the first name, a space, the middle name, and a right parenthesis.
LastSpaceFirst	Indicates that the contact is displayed as the last name, followed by a space, the first name, a space, and the middle name.
DisplayName<6>	Indicates that the contact is displayed as the display name.
FirstName<7>	Indicates that the contact is displayed as the first name.
LastFirstMiddleSuffix<8>	Indicates that the contact is displayed as the last name, the first name, the middle name, and the suffix for the contact.
LastName<9>	Indicates that the contact is displayed as the last name.
Empty<10>	Indicates that the contact is displayed as empty.

3.1.4.1.2.3 t:ImAddressKeyType Simple Type

The **ImAddressKeyType** enumeration represents the instant messaging addresses for a contact.

```
<xs:simpleType name="ImAddressKeyType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="ImAddress1"
     />
    <xs:enumeration
      value="ImAddress2"
     />
    <xs:enumeration
      value="ImAddress3"
     />
  </xs:restriction>
</xs:simpleType>
```

Enumeration

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

Value	Meaning
ImAddress1	Identifies the first instant messaging address for the user.

Value	Meaning
ImAddress2	Identifies the second instant messaging address for the user.
ImAddress3	Identifies the third instant messaging address for the user.

3.1.4.1.2.4 t:PhoneNumberKeyType Simple Type

The **PhoneNumberKeyType** simple type represents types of telephone numbers for a contact.

```
<xs:simpleType name="PhoneNumberKeyType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="AssistantPhone"
     />
    <xs:enumeration
      value="BusinessFax"
     />
    <xs:enumeration
      value="BusinessPhone"
     />
    <xs:enumeration
      value="BusinessPhone2"
     />
    <xs:enumeration
      value="Callback"
     />
    <xs:enumeration
      value="CarPhone"
     />
    <xs:enumeration
      value="CompanyMainPhone"
     />
    <xs:enumeration
      value="HomeFax"
     />
    <xs:enumeration
      value="HomePhone"
     />
    <xs:enumeration
      value="HomePhone2"
     />
    <xs:enumeration
      value="Isdn"
     />
    <xs:enumeration
      value="MobilePhone"
     />
    <xs:enumeration
      value="OtherFax"
     />
    <xs:enumeration
      value="OtherTelephone"
     />
    <xs:enumeration
      value="Pager"
```

```
      />
     <xs:enumeration
       value="PrimaryPhone"
      />
     <xs:enumeration
       value="RadioPhone"
      />
     <xs:enumeration
       value="Telex"
      />
     <xs:enumeration
       value="TtyTddPhone"
      />
   </xs:restriction>
 </xs:simpleType>
```

Enumeration

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

Value	Meaning
AssistantPhone	Identifies the telephone number as the assistant's telephone number.
BusinessFax	Identifies the telephone number as a business fax number.
BusinessPhone	Identifies the telephone number as a business telephone number.
BusinessPhone2	Identifies the telephone number as a second business telephone number.
Callback	Identifies the telephone number as a callback number.
CarPhone	Identifies the telephone number as a car telephone number.
CompanyMainPhone	Identifies the telephone number as the company's main telephone number.
HomeFax	Identifies the telephone number as a home fax number.
HomePhone	Identifies the telephone number as a home telephone number.
HomePhone2	Identifies the telephone number as a second home telephone number.
Isdn	Identifies the telephone number as an Integrated Services Digital Network (ISDN) line.
MobilePhone	Identifies the telephone number as a mobile phone number.
OtherFax	Identifies the telephone number as another fax number.
OtherTelephone	Identifies the telephone number as another telephone number.
Pager	Identifies the telephone number as a pager.
PrimaryPhone	Identifies the telephone number as the primary telephone number.
RadioPhone	Identifies the telephone number as a radio telephone.
Telex	Identifies the telephone number as a telex telephone number.

Value	Meaning
TtyTddPhone	Identifies the telephone number as a teletype/telecommunication device for the deaf (TTY/TDD) telephone number.

3.1.4.1.2.5 t:PhysicalAddressIndexType Simple Type

The **PhysicalAddressIndexType** simple type identifies the display types for physical addresses.

```
<xs:simpleType name="PhysicalAddressIndexType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="None"
     />
    <xs:enumeration
      value="Business"
     />
    <xs:enumeration
      value="Home"
     />
    <xs:enumeration
      value="Other"
     />
  </xs:restriction>
</xs:simpleType>
```

Enumeration

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

Value	Meaning
None	Indicates that no type is specified for the address.
Business	Displays the address as a business address.
Home	Displays the address as a home address.
Other	Displays the address as an address of type other.

3.1.4.1.2.6 t:PhysicalAddressKeyType Simple Type

The **PhysicalAddressKeyType** simple type identifies the types of physical addresses for a contact.

```
<xs:simpleType name="PhysicalAddressKeyType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="Business"
     />
    <xs:enumeration
      value="Home"
```

```
      />
     <xs:enumeration
       value="Other"
      />
   </xs:restriction>
 </xs:simpleType>
```

Enumeration

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

Value	Meaning
Business	Identifies the address as a business address.
Home	Identifies the address as a home address.
Other	Identifies the address as an address of type other.

3.1.4.2 DeleteItem

This protocol uses the **DeleteItem** operation specified in [MS-OXWSCORE] section 3.1.4.3 to delete contact item elements.

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

Request

Message format	Description
tns:DeleteItemSoapIn ([MS-OXWSCORE] section 3.1.4.3.1.1)	Specifies the SOAP message that defines the contact item to delete. The **DeleteItem** ([MS-OXWSCORE] section 3.1.4.3.2.1) child element that specifies the XML request MUST contain one or more **t:ItemIdType** ([MS-OXWSCORE] section 2.2.4.23) elements. All other elements MUST be empty.

Response

Message format	Description
tns:DeleteItemSoapOut ([MS-OXWSCORE] section 3.1.4.3.1.2)	Specifies the SOAP message returned by the server in response.

3.1.4.3 UpdateItem

This protocol uses the **UpdateItem** operation specified in [MS-OXWSCORE] section 3.1.4.9 to update contact item elements.

```
<wsdl:operation name="UpdateItem">
```

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

Request

Message format	Description
tns:UpdateItemSoapIn ([MS-OXWSCORE] section 3.1.4.9.1.1)	Specifies the SOAP message that defines the contact item to update. The **Items** child element of the **UpdateItem** ([MS-OXWSCORE] section 3.1.4.9.2.1) child element that specifies the XML request MUST contain one or more **t:ContactItemType** elements (section 2.2.4.2). All other elements MUST be empty.

Response

Message format	Description
tns:UpdateItemSoapOut ([MS-OXWSCORE] section 3.1.4.9.1.2)	Specifies the SOAP message returned by the server in response.

3.1.4.4 MoveItem

This protocol uses the **MoveItem** operation specified in [MS-OXWSCORE] section 3.1.4.7 to move contact item elements.

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

Request

Message format	Description
tns:MoveItemSoapIn ([MS-OXWSCORE] section 3.1.4.7.1.1)	Specifies the SOAP message that defines the contact item to move. The **Items** child element of the **MoveItem** child element ([MS-OXWSCORE] section 3.1.4.7.2.1) that specifies the XML request MUST contain the following elements: **t:TargetFolderIdType** ([MS-OXWSFOLD] section 2.2.4.14), and **t:ItemIdType** ([MS-OXWSCORE] section 2.2.4.23). All other elements MUST be empty.

Response

Message format	Description
tns:MoveItemSoapOut ([MS-OXWSCORE] section 3.1.4.7.1.2)	Specifies the SOAP message returned by the server in response.

3.1.4.5 CopyItem

This protocol uses the **CopyItem** operation specified in [MS-OXWSCORE] section 3.1.4.1 to copy contact item elements.

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

Request

Message format	Description
tns:CopyItemSoapIn ([MS-OXWSCORE] section 3.1.4.1.1.1)	Specifies the SOAP message that defines the contact item to copy. The **Items** child element of the **CopyItem** ([MS-OXWSCORE] section 3.1.4.1.2.1) child element that specifies the XML request MUST contain the following elements: **t:TargetFolderIdType** ([MS-OXWSFOLD] section 2.2.4.14), and **t:ItemIdType** ([MS-OXWSCORE] section 2.2.4.23). All other elements MUST be empty.

Response

Message format	Description
tns:CopyItemSoapOut ([MS-OXWSCORE] section 3.1.4.1.1.2)	Specifies the SOAP message returned by the server in response.

3.1.4.6 CreateItem

This protocol uses the **CreateItem** operation specified in [MS-OXWSCORE] section 3.1.4.2 to create contact item elements.

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

Request

Message format	Description
tns:CreateItemSoapIn ([MS-OXWSCORE] section 3.1.4.2.1.1)	Specifies the SOAP message that defines the contact item to create. The **Items** child element of the **CreateItem** ([MS-OXWSCORE] section 3.1.4.2.2.1) child element that specifies the XML request MUST contain one or more **t:ContactItemType** elements (section 2.2.4.2). All other elements MUST be empty.

Response

Message format	Description
tns:CreateItemSoapOut ([MS-OXWSCORE] section	Specifies the SOAP message returned by the

Message format	Description
3.1.4.2.1.2)	server in response.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

4 Protocol Examples

None.

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 to implement the functionality that is specified in this document.

File name	Description	Section
MS-OXWSCONT.wsdl	Contains the WSDL for the implementation of this protocol.	6
MS-OXWSCONT-types.xsd	Contains the XML schema type definitions that are used in this protocol.	7
MS-OXWSCORE-messages.xsd	Contains XML schema message definitions that are referred to by this protocol.	[MS-OXWSCORE] section 7.1

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

This section contains the content of the MS-OXWSCONT.wsdl file.

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:s="http://www.w3.org/2001/XMLSchema" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages">
     <wsdl:types>
          <xs:schema id="messages" elementFormDefault="qualified" version="Exchange2013"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages">
               <xs:include schemaLocation="MS-OXWSCORE-messages.xsd"/>
          </xs:schema>
     </wsdl:types>
     <wsdl:portType name="ExchangeServicePortType">
          <wsdl:operation name="GetItem">
               <wsdl:input message="tns:GetItemSoapIn"/>
               <wsdl:output message="tns:GetItemSoapOut"/>
          </wsdl:operation>
          <wsdl:operation name="CreateItem">
               <wsdl:input message="tns:CreateItemSoapIn"/>
               <wsdl:output message="tns:CreateItemSoapOut"/>
          </wsdl:operation>
          <wsdl:operation name="DeleteItem">
               <wsdl:input message="tns:DeleteItemSoapIn"/>
               <wsdl:output message="tns:DeleteItemSoapOut"/>
          </wsdl:operation>
          <wsdl:operation name="UpdateItem">
               <wsdl:input message="tns:UpdateItemSoapIn"/>
               <wsdl:output message="tns:UpdateItemSoapOut"/>
          </wsdl:operation>
          <wsdl:operation name="MoveItem">
               <wsdl:input message="tns:MoveItemSoapIn"/>
               <wsdl:output message="tns:MoveItemSoapOut"/>
          </wsdl:operation>
          <wsdl:operation name="CopyItem">
               <wsdl:input message="tns:CopyItemSoapIn"/>
```

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

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

```
                    <soap:header message="tns:CopyItemSoapOut" part="ServerVersion"
use="literal"/>
               </wsdl:output>
          </wsdl:operation>
     </wsdl:binding>
     <wsdl:message name="GetItemSoapIn">
          <wsdl:part name="request" element="tns:GetItem"/>
          <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
          <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
          <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
          <wsdl:part name="TimeZoneContext" element="t:TimeZoneContext"/>
     </wsdl:message>
     <wsdl:message name="GetItemSoapOut">
          <wsdl:part name="GetItemResult" element="tns:GetItemResponse"/>
          <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
     </wsdl:message>
     <wsdl:message name="CreateItemSoapIn">
          <wsdl:part name="request" element="tns:CreateItem"/>
          <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
          <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
          <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
          <wsdl:part name="TimeZoneContext" element="t:TimeZoneContext"/>
     </wsdl:message>
     <wsdl:message name="CreateItemSoapOut">
          <wsdl:part name="CreateItemResult" element="tns:CreateItemResponse"/>
          <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
     </wsdl:message>
     <wsdl:message name="DeleteItemSoapIn">
          <wsdl:part name="request" element="tns:DeleteItem"/>
          <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
          <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
          <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
     </wsdl:message>
     <wsdl:message name="DeleteItemSoapOut">
          <wsdl:part name="DeleteItemResult" element="tns:DeleteItemResponse"/>
          <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
     </wsdl:message>
     <wsdl:message name="UpdateItemSoapIn">
          <wsdl:part name="request" element="tns:UpdateItem"/>
          <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
          <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
          <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
          <wsdl:part name="TimeZoneContext" element="t:TimeZoneContext"/>
     </wsdl:message>
     <wsdl:message name="UpdateItemSoapOut">
          <wsdl:part name="UpdateItemResult" element="tns:UpdateItemResponse"/>
          <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
     </wsdl:message>
     <wsdl:message name="MoveItemSoapIn">
          <wsdl:part name="request" element="tns:MoveItem"/>
          <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
          <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
          <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
     </wsdl:message>
     <wsdl:message name="MoveItemSoapOut">
          <wsdl:part name="MoveItemResult" element="tns:MoveItemResponse"/>
          <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
     </wsdl:message>
     <wsdl:message name="CopyItemSoapIn">
```

```
            <wsdl:part name="request" element="tns:CopyItem"/>
            <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
            <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
            <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
      </wsdl:message>
      <wsdl:message name="CopyItemSoapOut">
            <wsdl:part name="CopyItemResult" element="tns:CopyItemResponse"/>
            <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
      </wsdl:message>
  </wsdl:definitions>
```

7 Appendix B: Full XML Schema

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

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

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

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

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types"
elementFormDefault="qualified" version="Exchange2013" id="types">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xs:include schemaLocation="MS-OXWSCDATA-types.xsd"/>
  <xs:complexType name="CompleteNameType">
    <xs:sequence>
      <xs:element name="Title" type="xs:string" minOccurs="0"/>
      <xs:element name="FirstName" type="xs:string" minOccurs="0"/>
      <xs:element name="MiddleName" type="xs:string" minOccurs="0"/>
      <xs:element name="LastName" type="xs:string" minOccurs="0"/>
      <xs:element name="Suffix" type="xs:string" minOccurs="0"/>
      <xs:element name="Initials" type="xs:string" minOccurs="0"/>
      <xs:element name="FullName" type="xs:string" minOccurs="0"/>
      <xs:element name="Nickname" type="xs:string" minOccurs="0"/>
      <xs:element name="YomiFirstName" type="xs:string" minOccurs="0"/>
      <xs:element name="YomiLastName" type="xs:string" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ContactItemType">
    <xs:complexContent>
      <xs:extension base="t:ItemType">
        <xs:sequence>
          <xs:element name="FileAs" type="xs:string" minOccurs="0"/>
          <xs:element name="FileAsMapping" type="t:FileAsMappingType" minOccurs="0"/>
          <xs:element name="DisplayName" type="xs:string" minOccurs="0"/>
          <xs:element name="GivenName" type="xs:string" minOccurs="0"/>
          <xs:element name="Initials" type="xs:string" minOccurs="0"/>
          <xs:element name="MiddleName" type="xs:string" minOccurs="0"/>
          <xs:element name="Nickname" type="xs:string" minOccurs="0"/>
          <xs:element name="CompleteName" type="t:CompleteNameType" minOccurs="0"/>
          <xs:element name="CompanyName" type="xs:string" minOccurs="0"/>
          <xs:element name="EmailAddresses" type="t:EmailAddressDictionaryType"
minOccurs="0"/>
          <xs:element name="PhysicalAddresses" type="t:PhysicalAddressDictionaryType"
minOccurs="0"/>
          <xs:element name="PhoneNumbers" type="t:PhoneNumberDictionaryType" minOccurs="0"/>
          <xs:element name="AssistantName" type="xs:string" minOccurs="0"/>
          <xs:element name="Birthday" type="xs:dateTime" minOccurs="0"/>
          <xs:element name="BusinessHomePage" type="xs:anyURI" minOccurs="0"/>
```

```
          <xs:element name="Children" type="t:ArrayOfStringsType" minOccurs="0"/>
          <xs:element name="Companies" type="t:ArrayOfStringsType" minOccurs="0"/>
          <xs:element name="ContactSource" type="t:ContactSourceType" minOccurs="0"/>
          <xs:element name="Department" type="xs:string" minOccurs="0"/>
          <xs:element name="Generation" type="xs:string" minOccurs="0"/>
          <xs:element name="ImAddresses" type="t:ImAddressDictionaryType" minOccurs="0"/>
          <xs:element name="JobTitle" type="xs:string" minOccurs="0"/>
          <xs:element name="Manager" type="xs:string" minOccurs="0"/>
          <xs:element name="Mileage" type="xs:string" minOccurs="0"/>
          <xs:element name="OfficeLocation" type="xs:string" minOccurs="0"/>
          <xs:element name="PostalAddressIndex" type="t:PhysicalAddressIndexType"
minOccurs="0"/>
          <xs:element name="Profession" type="xs:string" minOccurs="0"/>
          <xs:element name="SpouseName" type="xs:string" minOccurs="0"/>
          <xs:element name="Surname" type="xs:string" minOccurs="0"/>
          <xs:element name="WeddingAnniversary" type="xs:dateTime" minOccurs="0"/>
          <xs:element name="HasPicture" type="xs:boolean" minOccurs="0"/>
          <xs:element name="PhoneticFullName" type="xs:string" minOccurs="0" />
          <xs:element name="PhoneticFirstName" type="xs:string" minOccurs="0" />
          <xs:element name="PhoneticLastName" type="xs:string" minOccurs="0" />
          <xs:element name="Alias" type="xs:string" minOccurs="0" />
          <xs:element name="Notes" type="xs:string" minOccurs="0" />
          <xs:element name="Photo" type="xs:base64Binary" minOccurs="0" />
          <xs:element name="UserSMIMECertificate" type="t:ArrayOfBinaryType" minOccurs="0" />
          <xs:element name="MSExchangeCertificate" type="t:ArrayOfBinaryType" minOccurs="0"
/>
          <xs:element name="DirectoryId" type="xs:string" minOccurs="0" />
          <xs:element name="ManagerMailbox" type="t:SingleRecipientType" minOccurs="0" />
          <xs:element name="DirectReports" type="t:ArrayOfRecipientsType" minOccurs="0" />
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="ArrayOfBinaryType">
        <xs:sequence>
            <xs:element name="Base64Binary" type="xs:base64Binary" minOccurs="0"
maxOccurs="unbounded" />
        </xs:sequence>
     </xs:complexType>
  <xs:complexType name="ContactsFolderType">
    <xs:complexContent>
      <xs:extension base="t:BaseFolderType">
        <xs:sequence>
          <xs:element name="SharingEffectiveRights" type="t:PermissionReadAccessType"
minOccurs="0"/>
          <xs:element name="PermissionSet" type="t:PermissionSetType" minOccurs="0"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:simpleType name="ContactSourceType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="ActiveDirectory"/>
      <xs:enumeration value="Store"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="ContactsViewType">
    <xs:complexContent>
      <xs:extension base="t:BasePagingType">
```

```
        <xs:attribute name="InitialName" type="xs:string" use="optional"/>
        <xs:attribute name="FinalName" type="xs:string" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="EmailAddressDictionaryEntryType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="Key" type="t:EmailAddressKeyType" use="required"/>
        <xs:attribute name="Name" type="xs:string" use="optional"/>
        <xs:attribute name="RoutingType" type="xs:string" use="optional"/>
        <xs:attribute name="MailboxType" type="t:MailboxTypeType" use="optional"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="EmailAddressDictionaryType">
    <xs:sequence>
      <xs:element name="Entry" type="t:EmailAddressDictionaryEntryType"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="EmailAddressKeyType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="EmailAddress1"/>
      <xs:enumeration value="EmailAddress2"/>
      <xs:enumeration value="EmailAddress3"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="FileAsMappingType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="None"/>
      <xs:enumeration value="LastCommaFirst"/>
      <xs:enumeration value="FirstSpaceLast"/>
      <xs:enumeration value="Company"/>
      <xs:enumeration value="LastCommaFirstCompany"/>
      <xs:enumeration value="CompanyLastFirst"/>
      <xs:enumeration value="LastFirst"/>
      <xs:enumeration value="LastFirstCompany"/>
      <xs:enumeration value="CompanyLastCommaFirst"/>
      <xs:enumeration value="LastFirstSuffix"/>
      <xs:enumeration value="LastSpaceFirstCompany"/>
      <xs:enumeration value="CompanyLastSpaceFirst"/>
      <xs:enumeration value="LastSpaceFirst"/>
      <xs:enumeration value="DisplayName"/>
      <xs:enumeration value="FirstName"/>
      <xs:enumeration value="LastFirstMiddleSuffix"/>
      <xs:enumeration value="LastName"/>
      <xs:enumeration value="Empty"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="ImAddressDictionaryEntryType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="Key" type="t:ImAddressKeyType" use="required"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="ImAddressDictionaryType">
    <xs:sequence>
```

```
      <xs:element name="Entry" type="t:ImAddressDictionaryEntryType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="ImAddressKeyType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="ImAddress1"/>
      <xs:enumeration value="ImAddress2"/>
      <xs:enumeration value="ImAddress3"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="PhoneNumberDictionaryEntryType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="Key" type="t:PhoneNumberKeyType" use="required"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="PhoneNumberDictionaryType">
    <xs:sequence>
      <xs:element name="Entry" type="t:PhoneNumberDictionaryEntryType"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="PhoneNumberKeyType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="AssistantPhone"/>
      <xs:enumeration value="BusinessFax"/>
      <xs:enumeration value="BusinessPhone"/>
      <xs:enumeration value="BusinessPhone2"/>
      <xs:enumeration value="Callback"/>
      <xs:enumeration value="CarPhone"/>
      <xs:enumeration value="CompanyMainPhone"/>
      <xs:enumeration value="HomeFax"/>
      <xs:enumeration value="HomePhone"/>
      <xs:enumeration value="HomePhone2"/>
      <xs:enumeration value="Isdn"/>
      <xs:enumeration value="MobilePhone"/>
      <xs:enumeration value="OtherFax"/>
      <xs:enumeration value="OtherTelephone"/>
      <xs:enumeration value="Pager"/>
      <xs:enumeration value="PrimaryPhone"/>
      <xs:enumeration value="RadioPhone"/>
      <xs:enumeration value="Telex"/>
      <xs:enumeration value="TtyTddPhone"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="PhysicalAddressDictionaryEntryType">
    <xs:sequence>
      <xs:element name="Street" type="xs:string" minOccurs="0"/>
      <xs:element name="City" type="xs:string" minOccurs="0"/>
      <xs:element name="State" type="xs:string" minOccurs="0"/>
      <xs:element name="CountryOrRegion" type="xs:string" minOccurs="0"/>
      <xs:element name="PostalCode" type="xs:string" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="Key" type="t:PhysicalAddressKeyType" use="required"/>
  </xs:complexType>
  <xs:complexType name="PhysicalAddressDictionaryType">
    <xs:sequence>
```

```
      <xs:element name="Entry" type="t:PhysicalAddressDictionaryEntryType"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="PhysicalAddressIndexType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="None"/>
      <xs:enumeration value="Business"/>
      <xs:enumeration value="Home"/>
      <xs:enumeration value="Other"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="PhysicalAddressKeyType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="Business"/>
      <xs:enumeration value="Home"/>
      <xs:enumeration value="Other"/>
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
```

8 Appendix C: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Microsoft Exchange Server 2007
- Microsoft Exchange Server 2010
- Microsoft Exchange Server 2013

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

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

<1> Section 2.2.4.1: Exchange 2007, Exchange 2010, and Exchange 2010 SP1 do not include the **ArrayOfBinaryType** complex type. This type was introduced in Exchange 2010 SP2.

<2> Section 2.2.4.2: Exchange 2007, Exchange 2010, and Exchange 2010 SP1 do not include the following elements: **PhoneticFullName**, **PhoneticFirstName**, **PhoneticLastName**, **Alias**, **Notes**, **Photo**, **UserSMIMECertificate**, **MSExchangeCertificate**, **DirectoryId**, **ManagerMailbox**, and **DirectReports**. These elements were introduced in Exchange 2010 SP2.

<3> Section 3.1.4.1.1.4: Exchange 2007 does not support the **Name** attribute.

<4> Section 3.1.4.1.1.4: Exchange 2007 does not support the **RoutingType** attribute.

<5> Section 3.1.4.1.1.4: Exchange 2007 does not support the **MailboxType** attribute.

<6> Section 3.1.4.1.2.2: Exchange 2007 does not support the **DisplayName** attribute.

<7> Section 3.1.4.1.2.2: Exchange 2007 does not support the **FirstName** attribute.

<8> Section 3.1.4.1.2.2: Exchange 2007 does not support the **LastFirstMiddleSuffix** attribute.

<9> Section 3.1.4.1.2.2: Exchange 2007 does not support the **LastName** attribute.

<10> Section 3.1.4.1.2.2: Exchange 2007 does not support the **Empty** attribute.

9 Change Tracking

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

10 Index

T

V

W

X