[MS-OXWSRSLNM]: Resolve Recipient Names 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.
- **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	1.2.1	No change	No changes to the meaning, language, or formatting of the technical content.
11/03/2010	2.0	Major	Significantly changed the technical content.

Contents

1 Introduction 5
1.1 Glossary 5
1.2 References 5
1.2.1 Normative References 5
1.2.2 Informative References 6
1.3 Overview 6
1.4 Relationship to Other Protocols 6
1.5 Prerequisities/Preconditions 6
1.6 Applicability Statement 7
1.7 Versioning and Capability Negotiation 7
1.8 Vendor-Extensible Fields 7
1.9 Standards Assignments 7

2 Messages 8
2.1 Transport 8
2.2 Common Message Syntax 8
2.2.1 Namespaces 8
2.2.2 Simple Types 8
2.2.3 Complex Types 8
2.2.4 Elements 8
2.2.5 Attributes 8
2.2.6 Groups 8
2.2.7 Attribute Groups 9
2.2.8 Message Syntax 9

3 Protocol Details 10
3.1 ExchangeServicePortType Server Details 10
3.1.1 Server Abstract Data Model 10
3.1.2 Server Timers 10
3.1.3 Server Initialization 10
3.1.4 Server Message Processing Events and Sequencing 10
3.1.4.1 ResolveNames 10
3.1.4.1.1 ResolveNames Simple Types 11
3.1.4.1.1.1 t:ResolveNamesSearchScopeType Simple Type 11
3.1.4.1.2 ResolveNames Complex Types 12
3.1.4.1.2.1 m:ResolveNamesResponseMessageType Complex Type 12
3.1.4.1.2.2 m:ResolveNamesResponseType Complex Type 12
3.1.4.1.2.3 m:ResolveNamesType Complex Type 13
3.1.4.1.2.4 t:ArrayOfResolutionType Complex Type 13
3.1.4.1.2.5 t:Resolutiontype Complex Type 14
3.1.4.1.3 ResolveNames Elements 15
3.1.4.1.3.1 ResolveNames Element 15
3.1.4.1.3.2 ResolveNamesResponse Element 15
3.1.4.1.4 ResolveNames Messages 15
3.1.4.1.4.1 tns:ResolveNamesSoapIn 15
3.1.4.1.4.2 tns:ResolveNamesSoapOut 15
3.1.5 Server Timer Events 16
3.1.6 Server Other Local Events 16
3.2 Client Details 16
3.2.1 Client Abstract Data Model 16

3.2.2 Client Timers 16
3.2.3 Client Initialization 16
3.2.4 Client Message Processing Events and Sequencing 16
3.2.5 Client Timer Events 16
3.2.6 Client Other Local Events 16

4 Protocol Examples 17

5 Security 18
5.1 Security Considerations for Implementers 18
5.2 Index of Security Parameters 18

6 Appendix A: Full WSDL 19
6.1 WSDL 19
6.2 Messages Schema 20
6.3 Types Schema 21

7 Appendix A: Product Behavior 23

8 Change Tracking 24

9 Index 26

1 Introduction

This document specifies the Resolve Recipient Names Web Service protocol, which enables a client that has incomplete recipient identifying information to retrieve a list of matching and similar recipients that are known to the server.

1.1 Glossary

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

Hypertext Transfer Protocol (HTTP)
Hypertext Transfer Protocol over Secure Socket Layers (HTTPS)
mailbox
SOAP body
SOAP fault
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:

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. 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

[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

The Resolve Recipient Names Web Service protocol enables a client to retrieve a list of e-mail addresses or contacts that are known to the server. The client can provide the full or partial name for the intended recipient. The server that is implementing the Resolve Recipient Names Web Service protocol will return a list of matching or similarly named recipients. The client can then use those names when addressing items for the server to send. The server can return valid recipients that are known to it; for example, recipient names that are stored in a directory service or in a global or user-specific list of contacts.

1.4 Relationship to Other Protocols

The Resolve Recipient Names Web Service protocol uses SOAP over **HTTP** and SOAP over **HTTPS**, as shown in the following figures.

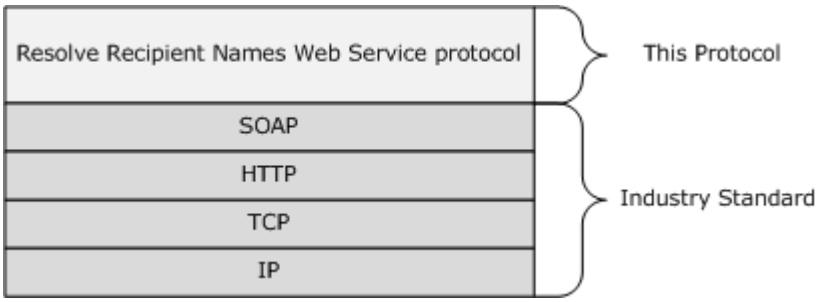

Figure 1: Resolve Recipient Names Web Service protocol HTTP stack

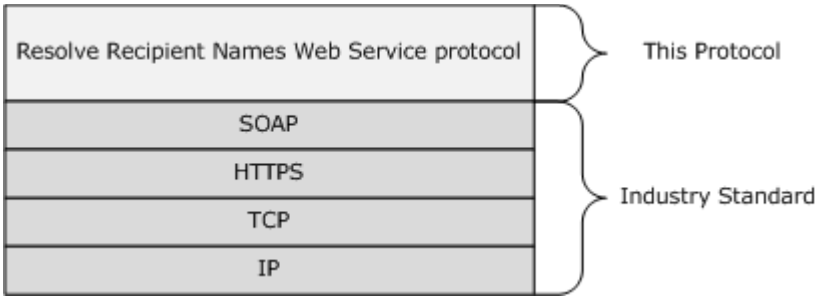

Figure 2: Resolve Recipient Names Web Service protocol HTTPS stack

1.5 Prerequisities/Preconditions

None.

1.6 Applicability Statement

The Resolve Recipient Names Web Service protocol is applicable to clients that must obtain a list of candidate recipients that are possible matches for ambiguous or partial names that are provided to the client. The server returns the list of candidate matches, and the client application determines whether any of the supplied candidates are the intended recipient.

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:** 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 includes text strings in various messages. Localization considerations for such strings are specified in section 3.1.4.
- **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 **Web Services Description Language (WSDL)** as defined in [WSDL].

2.2.1 Namespaces

This specification defines and references various **XML namespaces** by using the mechanisms specified in [XMLNS]. Although this specification associates a specific XML namespace prefix with each XML namespace that is used, the choice of any particular XML namespace prefix is implementation-specific and is not significant for interoperability.

Prefix	Namespace URI	Reference
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
tns	http://schemas.microsoft.com/exchange/services/2006/messages	[MS-OXWSRSLNM]
s	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1]
targetNamespace	http://schemas.microsoft.com/exchange/services/2006/messages	[MS-OXWSRSLNM]
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
t	http://schemas.microsoft.com/exchange/services/2006/types	[MS-OXWSRSLNM]

2.2.2 Simple Types

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

2.2.3 Complex Types

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

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 Message Syntax

This specification does not define any common XML schema message syntax 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

The Resolve Recipient Names Web Service protocol defines a single port type with one operation.

Operation	Description
ResolveNames	Returns a list of candidate recipient names that are known to the server that match or are similar to the recipient name provided in the request by the client. The **ResolveNames** operation is defined in section 3.1.4.1.

3.1.1 Server Abstract Data Model

The Resolve Recipient Names Web service protocol is a stateless protocol.

3.1.2 Server Timers

None.

3.1.3 Server Initialization

None.

3.1.4 Server Message Processing Events and Sequencing

This protocol includes the operation listed in the following table.

Operation	Description
ResolveNames	Returns a list of candidate recipient names that are known to the server that match or are similar to the recipient name provided in the request by the client.

3.1.4.1 ResolveNames

The **ResolveNames** operation returns a list of candidate recipients that match an ambiguous recipient name that is supplied by the client.

```
<wsdl:operation name="ResolveNames" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
    <wsdl:input message="tns:ResolveNamesSoapIn"/>
    <wsdl:output message="tns:ResolveNamesSoapOut"/>
</wsdl:operation>
```

Request

Message Format	Description
tns:ResolveNamesSoapIn	Specifies the **SOAP message** that requests a list of candidate names.

Response

Message Format	Description
tns:ResolveNamesSoapOut	Specifies the SOAP message that is returned by the server in response.

3.1.4.1.1 ResolveNames Simple Types

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

3.1.4.1.1.1 t:ResolveNamesSearchScopeType Simple Type

The **ResolveNamesSearchScopeType** simple type specifies a location where the server searches for candidate matches.

```
<xs:simpleType name="ResolveNamesSearchScopeType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="ActiveDirectory"
     />
    <xs:enumeration
      value="ActiveDirectoryContacts"
     />
    <xs:enumeration
      value="Contacts"
     />
    <xs:enumeration
      value="ContactsActiveDirectory"
     />
  </xs:restriction>
</xs:simpleType>
```

Enumeration

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

Value	Description
ActiveDirectory	Specifies that the server searches for candidate matches in the directory service for the organization.
ActiveDirectoryContacts	Specifies that the server searches for candidate matches in the global contacts list that is stored in the directory service for the organization.
Contacts	Specifies that the server searches for candidate matches in the **mailbox** Contacts folder.

Value	Description
ContactsActiveDirectory	Specifies that the server searches for candidate matches in the mailbox Contacts list that is stored in the directory service for the organization.

3.1.4.1.2 ResolveNames Complex Types

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

3.1.4.1.2.1 m:ResolveNamesResponseMessageType Complex Type

The **ResolveNamesResponseMessageType** complex type specifies the status and result of a **ResolveNames** operation (section 3.1.4.1). The **ResolveNamesResponseMessageType** complex type extends the **ResponseMessageType** complex type, as specified in [MS-OXWSCDATA] section 2.2.3.52.

```
<xs:complexType name="ResolveNamesResponseMessageType">
  <xs:complexContent>
    <xs:extension
      base="m:ResponseMessageType"
    >
      <xs:sequence>
        <xs:element name="ResolutionSet"
          type="t:ArrayOfResolutionType"
          minOccurs="0"
         />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Child Elements

Element	Type
ResolutionSet	**t:ArrayOfResolutionType**

3.1.4.1.2.2 m:ResolveNamesResponseType Complex Type

The **ResolveNamesResponseType** complex type specifies the contents of the response from the server. The **ResolveNamesResponseType** complex type extends the **BaseResponseMessageType** complex type, as specified in [MS-OXWSCDATA] section 2.2.3.15.

```
<xs:complexType name="ResolveNamesResponseType">
  <xs:complexContent>
    <xs:extension
      base="m:BaseResponseMessageType"
     />
  </xs:complexContent>
</xs:complexType>
```

3.1.4.1.2.3 m:ResolveNamesType Complex Type

The **ResolveNamesType** complex type specifies the content of a request from the client to locate candidate matching recipients. The **ResolveNamesType** complex type extends the **BaseRequestType** complex type, as specified in [MS-OXWSCDATA] section 2.2.3.52.

```
<xs:complexType name="ResolveNamesType">
  <xs:complexContent>
    <xs:extension
      base="m:BaseRequestType"
    >
      <xs:sequence>
        <xs:element name="ParentFolderIds"
          type="t:NonEmptyArrayOfBaseFolderIdsType"
          minOccurs="0"
         />
        <xs:element name="UnresolvedEntry"
          type="t:NonEmptyStringType"
         />
      </xs:sequence>
      <xs:attribute name="ReturnFullContactData"
        type="xs:boolean"
        use="required"
       />
      <xs:attribute name="SearchScope"
        type="t:ResolveNamesSearchScopeType"
        default="ActiveDirectoryContacts"
       />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Child Elements

Element	Type
ParentFolderIds	t:NonEmptyArrayOfBaseFolderIdsType
UnresolvedEntry	t:NonEmptyStringType

Attributes

Name	Type
ReturnFullContactData	xs:boolean
SearchScope	t:ResolveNamesSearchScopeType

3.1.4.1.2.4 t:ArrayOfResolutionType Complex Type

The **ArrayOfResolutionType** complex type specifies the list of candidate matching recipients that are returned by the server. The **ArrayOfResolutionType** complex type does not extend any other complex type. The server does not return more than 100 candidate matches.

```
<xs:complexType name="ArrayOfResolutionType">
  <xs:sequence>
    <xs:element name="Resolution"
      type="t:ResolutionType"
      maxOccurs="100"
      minOccurs="0"
     />
  </xs:sequence>
  <xs:attributeGroup
    ref="t:FindResponsePagingAttributes"
   />
</xs:complexType>
```

Child Elements

Element	Type
Resolution	t:ResolutionType

Attribute Groups

Name
t:FindResponsePagingAttributes

3.1.4.1.2.5 t:Resolutiontype Complex Type

The **ResolutionType** complex type specifies the type of candidate matching recipient that is returned by the server. The **ResolutionType** complex type does not extend any other complex type.

```
<xs:complexType name="ResolutionType">
  <xs:sequence>
    <xs:element name="Mailbox"
      type="t:EmailAddressType"
     />
    <xs:element name="Contact"
      type="t:ContactItemType"
      minOccurs="0"
     />
  </xs:sequence>
</xs:complexType>
```

Child Elements

Element	Type	Description
Mailbox	t:EmailAddressType	Represents an e-mail address.
Contact	t:ContactItemType	Represents a contact item.

3.1.4.1.3 ResolveNames Elements

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

3.1.4.1.3.1 ResolveNames Element

The <ResolveNames> element is a container that specifies the parameters and the unresolved recipient name to the server.

```
<xs:element name="ResolveNames"
  type="m:ResolveNamesType"
 />
```

3.1.4.1.3.2 ResolveNamesResponse Element

The <ResolveNamesResponse> element is a container that specifies the candidate matches that are returned in the server response.

```
<xs:element name="ResolveNamesResponse"
  type="m:ResolveNamesResponseType"
 />
```

3.1.4.1.4 ResolveNames Messages

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

3.1.4.1.4.1 tns:ResolveNamesSoapIn

The **ResolveNamesSoapIn** message contains four parts, as described in the following table.

Part Name	Element/Type	Description
request	**tns:ResolveNames**	Specifies the request.
Impersonation	**t:ExchangeImpersonation**	The identifier of the account to impersonate.
MailboxCulture	**t:MailboxCulture**	Specifies the culture to use for accessing the server. The cultures are defined by [RFC3066].
RequestVersion	**t:RequestServerVersion**	Specifies the schema version for the request.

3.1.4.1.4.2 tns:ResolveNamesSoapOut

The **ResolveNamesSoapOut** message contains two parts, as described in the following table.

Part Name	Element/Type	Description
ResolveNamesResult	**tns:ResolveNamesResponse**	Specifies the response.
ServerVersion	**t:ServerVersionInfo**	Specifies the server version for the response.

3.1.5 Server Timer Events

None.

3.1.6 Server Other Local Events

None.

3.2 Client 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.

3.2.1 Client Abstract Data Model

None.

3.2.2 Client Timers

None.

3.2.3 Client Initialization

None.

3.2.4 Client Message Processing Events and Sequencing

None.

3.2.5 Client Timer Events

None.

3.2.6 Client Other Local Events

None.

4 Protocol Examples

None.

5 Security

5.1 Security Considerations for Implementers

The Resolve Recipient Names 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-OXWSRSLNM.wsdl	Contains the WSDL for the implementation of this protocol.	6.1
MS-OXWSRSLNM-messages.xsd	Contains the XML schema message definitions that are used in this protocol.	6.2
MS-OXWSRSLNM-types.xsd	Contains the XML schema type definitions that are used in this protocol.	6.3

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

6.1 WSDL

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

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

File name	Defining specification
MS-OXWSRSLNM-messages.xsd	[MS-OXWSRSLNM] section 6.2.
MS-OXWSRSLNM-types.xsd	[MS-OXWSRSLNM] section 6.3

```
<?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"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
<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">
<xs:include schemaLocation="MS-OXWSRSLNM-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:tns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:include schemaLocation="MS-OXWSRSLNM-types.xsd"/>
```

```
</xs:schema>
</wsdl:types>
<wsdl:portType name="ExchangeServicePortType">
<wsdl:operation name="ResolveNames" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
<wsdl:input message="tns:ResolveNamesSoapIn"/>
<wsdl:output message="tns:ResolveNamesSoapOut"/>
</wsdl:operation>
</wsdl:portType>
<wsdl:message name="ResolveNamesSoapIn" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
<wsdl:part name="request" element="tns:ResolveNames"/>
<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="ResolveNamesSoapOut" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
<wsdl:part name="ResolveNamesResult" element="tns:ResolveNamesResponse"/>
<wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
<wsdl:binding name="ExchangeServiceBinding" type="tns:ExchangeServicePortType">
<soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"/>
<wsdl:operation name="ResolveNames" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
<soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/ResolveNames"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"/>
<wsdl:input>
<soap:body parts="request" use="literal" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"/>
<soap:header message="tns:ResolveNamesSoapIn" part="Impersonation" use="literal"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"/>
<soap:header message="tns:ResolveNamesSoapIn" part="MailboxCulture" use="literal"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"/>
<soap:header message="tns:ResolveNamesSoapIn" part="RequestVersion" use="literal"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"/>
</wsdl:input>
<wsdl:output>
<soap:body parts="ResolveNamesResult" use="literal"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"/>
<soap:header message="tns:ResolveNamesSoapOut" part="ServerVersion" use="literal"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"/>
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
</wsdl:definitions>
```

6.2 Messages Schema

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

MS-OXWSRSLNM-messages.xsd includes or imports the files listed in the following table. For the schema file to operate correctly, these files have 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	[MS-OXWSCDATA] section 6.2
MS-OXWSRSLNM-types.xsd	[MS-OXWSRSLNM] section 6.3

```
<?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:import namespace="http://schemas.microsoft.com/exchange/services/2006/types"
schemaLocation="MS-OXWSRSLNM-types.xsd"/>
<xs:include schemaLocation="MS-OXWSCDATA-messages.xsd"/>
<xs:complexType name="ResolveNamesResponseMessageType"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:complexContent>
<xs:extension base="m:ResponseMessageType">
<xs:sequence>
<xs:element name="ResolutionSet" type="t:ArrayOfResolutionType" minOccurs="0"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="ResolveNamesResponseType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:complexContent>
<xs:extension base="m:BaseResponseMessageType"/>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="ResolveNamesType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:complexContent>
<xs:extension base="m:BaseRequestType">
<xs:sequence>
<xs:element name="ParentFolderIds" type="t:NonEmptyArrayOfBaseFolderIdsType" minOccurs="0"/>
<xs:element name="UnresolvedEntry" type="t:NonEmptyStringType"/>
</xs:sequence>
<xs:attribute name="ReturnFullContactData" type="xs:boolean" use="required"/>
<xs:attribute name="SearchScope" type="t:ResolveNamesSearchScopeType"
default="ActiveDirectoryContacts"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:element name="ResolveNames" type="m:ResolveNamesType"
xmlns:xs="http://www.w3.org/2001/XMLSchema"/>
<xs:element name="ResolveNamesResponse" type="m:ResolveNamesResponseType"
xmlns:xs="http://www.w3.org/2001/XMLSchema"/>
</xs:schema>
```

6.3 Types Schema

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

MS-OXWSRSLNM-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-types.xsd	[MS-OXWSCDATA] 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="ArrayOfResolutionType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:sequence>
<xs:element name="Resolution" type="t:ResolutionType" minOccurs="0" maxOccurs="100"/>
</xs:sequence>
<xs:attributeGroup ref="t:FindResponsePagingAttributes"/>
</xs:complexType>
<xs:complexType name="ResolutionType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:sequence>
<xs:element name="Mailbox" type="t:EmailAddressType"/>
<xs:element name="Contact" type="t:ContactItemType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="ResolveNamesSearchScopeType"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:restriction base="xs:string">
<xs:enumeration value="ActiveDirectory"/>
<xs:enumeration value="ActiveDirectoryContacts"/>
<xs:enumeration value="Contacts"/>
<xs:enumeration value="ContactsActiveDirectory"/>
</xs:restriction>
</xs:simpleType>
</xs:schema>
```

7 Appendix A: Product Behavior

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

- Microsoft® Exchange Server 2007
- Microsoft® Exchange Server 2010

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.

8 Change Tracking

This section identifies changes that were made to the [MS-OXWSRSLNM] 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.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change Type
1.2.1 Normative References	Added reference to [RFC3066].	N	Content updated.
7 Appendix A: Product Behavior	57840 Added Exchange 2007 to the list of applicable products.	Y	Content updated.

9 Index

A

Applicability 7

C

Capability negotiation 7
Change tracking 24
Client
  abstract data model 16
  initialization 16
  local events 16
  message processing 16
  overview 16
  sequencing rules 16
  timer events 16
  timers 16

F

Full WSDL 19

G

Glossary 5

I

Introduction 5

M

Messages
  overview 8
  syntax 8
  transport 8

O

Overview (synopsis) 6

P

Preconditions 6
Prerequisites 6
Product behavior 23

R

References
  informative 6
  normative 5
Relationship to other protocols 6

S

Security
  implementer considerations 18
  overview 18
  parameter index 18
Server
  abstract data model 10
  initialization 10
  local events 16
  message processing 10
  overview 10
  sequencing rules 10
  timer events 16
  timers 10
Standards assignments 7

T

Tracking changes 24

V

Vendor-extensible fields 7
Versioning 7