# [MS-OXOABK]:
# Address Book Object Protocol Specification

**Intellectual Property Rights Notice for Open Specifications Documentation**

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.

- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.

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

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

- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.

- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

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

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

**Preliminary Documentation.** This Open Specification provides documentation for past and current releases and/or for the pre-release (beta) version of this technology.  This Open Specification is final

documentation for past or current releases as specifically noted in the document, as applicable; it is preliminary documentation for the pre-release (beta) versions.  Microsoft will release final documentation in connection with the commercial release of the updated or new version of this technology.  As the documentation may change between this preliminary version and the final version of this technology, there are risks in relying on preliminary documentation. To the extent that you incur additional development obligations or any other costs as a result of relying on this preliminary documentation, you do so at your own risk.

## Revision Summary

| Date | Revision History | Revision Class | Comments |
|------|------------------|----------------|----------|
| 04/04/2008 | 0.1 | Major | Initial Availability. |
| 04/25/2008 | 0.2 | Minor | Revised and updated property names and other technical content. |
| 06/27/2008 | 1.0 | Major | Initial Release. |
| 08/06/2008 | 1.0.1 | Editorial | Updated references to reflect date of initial release. |
| 09/03/2008 | 1.0.2 | Editorial | Revised and edited technical content. |
| 12/03/2008 | 1.0.3 | Editorial | Revised and edited technical content. |
| 03/04/2009 | 1.0.4 | Editorial | Revised and edited technical content. |
| 04/10/2009 | 2.0 | Major | Updated technical content and applicable product releases. |
| 07/15/2009 | 3.0 | Major | Revised and edited for technical content. |
| 11/04/2009 | 3.1.0 | Minor | Updated the technical content. |
| 02/10/2010 | 3.2.0 | Minor | Updated the technical content. |
| 05/05/2010 | 4.0.0 | Major | Updated and revised the technical content. |
| 08/04/2010 | 4.1 | Minor | Clarified the meaning of the technical content. |
| 11/03/2010 | 4.2 | Minor | Clarified the meaning of the technical content. |
| 03/18/2011 | 5.0 | Major | Significantly changed the technical content. |
| 08/05/2011 | 6.0 | Major | Significantly changed the technical content. |
| 10/07/2011 | 6.1 | Minor | Clarified the meaning of the technical content. |
| 01/20/2012 | 7.0 | Major | Significantly changed the technical content. |
| 04/27/2012 | 8.0 | Major | Significantly changed the technical content. |
| 07/16/2012 | 8.0 | No change | No changes to the meaning, language, or formatting of the technical content. |

# Table of Contents

5 / 73

# 1 Introduction

The Address Book Object Protocol provides implementers with information about all the properties defined on various Address Book objects and how the properties on **Address Book objects** interrelate with one another.

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]:

**ambiguous name resolution (ANR)**
**ASCII**
**attribute**
**Augmented Backus-Naur Form (ABNF)**
**code page**
**common name (CN)**
**Coordinated Universal Time (UTC)**
**distinguished name (DN)**
**flags**
**GUID**
**little-endian**
**name service provider interface (NSPI)**
**relative distinguished name (RDN)**
**remote procedure call (RPC)**
**Unicode**

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

**address book**
**address book container**
**address book hierarchy table**
**Address Book object**
**address list**
**address type**
**alias**
**binary large object (BLOB)**
**contact**
**Department object**
**departmental group**
**display template**
**distribution list**
**entry ID**
**File Transfer Protocol (FTP)**
**Global Address List (GAL)**
**hierarchy table**
**Integrated Services Digital Network (ISDN)**
**locale**
**mail tip**
**mail user**
**mailbox**

**meeting request**
**messaging object**
**Minimal Entry ID**
**Multipurpose Internet Mail Extensions (MIME)**
**named property**
**offline address book (OAB)**
**Organization object**
**Permanent Entry ID**
**property ID**
**property tag**
**property type**
**recipient**
**Recipient object**
**Resource object**
**Rich Text Format (RTF)**
**S/MIME (Secure/Multipurpose Internet Mail Extensions)**
**Simple Mail Transfer Protocol (SMTP)**
**store**
**Transport Neutral Encapsulation Format (TNEF)**
**Uniform Resource Locator (URL)**
**UTF-16LE**

The following terms are specific to this document:

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

## 1.2 References

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

### 1.2.1 Normative References

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

[ASF] Microsoft Corporation, "Advanced Systems Format Specification", December 2004,
http://download.microsoft.com/download/7/9/0/790fecaa-f64a-4a5e-a430-
0bccdab3f1b4/ASF_Specification.doc

If you have any trouble finding [ASF], please check here.

[MS-NSPI] Microsoft Corporation, "Name Service Provider Interface (NSPI) Protocol Specification".

[MS-OXABREF] Microsoft Corporation, "Address Book Name Service Provider Interface (NSPI)
Referral Protocol Specification".

[MS-OXCDATA] Microsoft Corporation, "Data Structures".

[MS-OXCFOLD] Microsoft Corporation, "Folder Object Protocol Specification".

[MS-OXCMAIL] Microsoft Corporation, "RFC2822 and MIME to E-Mail Object Conversion Algorithm".

[MS-OXCMSG] Microsoft Corporation, "Message and Attachment Object Protocol Specification".

[MS-OXCPERM] Microsoft Corporation, "Exchange Access and Operation Permissions Protocol Specification".

[MS-OXCPRPT] Microsoft Corporation, "Property and Stream Object Protocol Specification".

[MS-OXCTABL] Microsoft Corporation, "Table Object Protocol Specification".

[MS-OXNSPI] Microsoft Corporation, "Exchange Server Name Service Provider Interface (NSPI) Protocol Specification".

[MS-OXOAB] Microsoft Corporation, "Offline Address Book (OAB) File Format and Schema".

[MS-OXOABKT] Microsoft Corporation, "Address Book User Interface Templates Protocol Specification".

[MS-OXOCNTC] Microsoft Corporation, "Contact Object Protocol Specification".

[MS-OXPFOAB] Microsoft Corporation, "Offline Address Book (OAB) Public Folder Retrieval Protocol Specification".

[MS-OXPROPS] Microsoft Corporation, "Exchange Server Protocols Master Property List".

[MS-OXWOAB] Microsoft Corporation, "Offline Address Book (OAB) Retrieval File Format".

[MS-RPCE] Microsoft Corporation, "Remote Procedure Call Protocol Extensions".

[RFC1034] Mockapetris, P., "Domain Names - Concepts and Facilities", STD 13, RFC 1034, November 1987, http://www.ietf.org/rfc/rfc1034.txt

[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

[RFC2822] Resnick, P., Ed., "Internet Message Format", STD 11, RFC 2822, April 2001, http://www.ietf.org/rfc/rfc2822.txt

[RFC2876] Pawling, J., "Use of the KEA and SKIPJACK Algorithms in CMS", RFC 2876, July 2000, http://www.rfc-editor.org/rfc/rfc2876.txt

[RFC3174] Eastlake III, D., and Jones, P., "US Secure Hash Algorithm 1 (SHA1)", RFC 3174, September 2001, http://www.ietf.org/rfc/rfc3174.txt

[RFC3280] Housley, R., Polk, W., Ford, W., and Solo, D., "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 3280, April 2002, http://www.ietf.org/rfc/rfc3280.txt

[RFC3852] Housley, R., "Cryptographic Message Syntax (CMS)", RFC 3852, July 2004, http://www.ietf.org/rfc/rfc3852.txt

[RFC5234] Crocker, D., Ed., and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008, http://www.rfc-editor.org/rfc/rfc5234.txt

[RFC5751] Ramsdell, B., and Turner, S., "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Message Specification", RFC 5751, January 2010, http://www.rfc-editor.org/rfc/rfc5751.txt

[RFC959] Postel, J., and Reynolds, J., "File Transfer Protocol (FTP)", RFC 959, October 1985, http://www.ietf.org/rfc/rfc959.txt

### 1.2.2 Informative References

[MS-GLOS] Microsoft Corporation, "Windows Protocols Master Glossary".

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

## 1.3 Overview

An **address book** is a collection of Address Book objects. There are many types of objects in an address book, many of which can be added as an addressee to an outbound message. Every object in the address book has various properties, and the values of these properties contain information that is used by messaging clients to route the message correctly or to display information about the Address Book object to a messaging user.

The address book also contains one or more **address book containers**, each of which describes an **address list**. An address list is a collection of Address Book objects that can be rendered in a table to be browsed by a messaging user. A collection of address book containers, each representing an address list, is arranged in an **address book hierarchy table**.

The address book can be stored remotely on an **NSPI** server and accessed through NSPI calls by using either of the NSPI protocols, as described in [MS-NSPI] and [MS-OXNSPI]. Alternatively, the address book can be stored locally on the client computer as an **offline address book (OAB)** by using the OAB Format and Schema Protocol, as described in [MS-OXOAB].

This protocol defines the properties on each of the Address Book objects and defines how they interrelate with one another.

## 1.4 Relationship to Other Protocols

This protocol relies on an understanding of how NSPI calls are transmitted to the server by using the underlying **remote procedure call (RPC)** transport, which uses the Name Service Provider Interface (NSPI) Protocol, as described in [MS-NSPI] or the Exchange Server NSPI Protocol, as described in [MS-OXNSPI], and an understanding of the OAB Format and Schema Protocol, as described in [MS-OXOAB].

The specification also relies on an understanding of how to locate and establish a connection with an NSPI server by using the Name Service Provider Interface (NSPI) Referral Protocol (as described in [MS-OXABREF]) and an understanding of how to obtain an OAB by using either the OAB Web Retrieval protocol (as described in [MS-OXWOAB]) or the OAB Public Folder Retrieval Protocol (as described in [MS-OXPFOAB]). The Address Book User Interface Templates Protocol, as described in [MS-OXOABKT], is needed to render information about an Address Book object to a messaging user, which relies on an understanding of this protocol.

## 1.5 Prerequisites/Preconditions

This specification assumes that either the messaging client has been referred to an NSPI server by using the NSPI Referral protocol, as specified in [MS-OXABREF], and has established a connection to a server that supports the NSPI Protocol or the Exchange Server NSPI Protocol, as specified in [MS-NSPI] and [MS-OXNSPI] respectively, or that the messaging client has access to an OAB, as specified in [MS-OXOAB].

## 1.6   Applicability Statement

This protocol is used to access information about Address Book objects in an organization. Messaging clients use this protocol to determine the destination for outbound messages that are addressed to these objects, and to display information about these objects to a messaging user.

## 1.7   Versioning and Capability Negotiation

None.

## 1.8   Vendor-Extensible Fields

None.

## 1.9   Standards Assignments

None.

# 2 Messages

The following sections specify the properties of Address Book objects and their formats.

Unless otherwise specified, all numeric values in this protocol are in **little-endian** format.

Unless otherwise specified, all **Unicode** string representations are in **UTF-16LE** format.

## 2.1 Transport

***Note*** *Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

This protocol uses either the Name Service Provider Interface (NSPI) Protocol, as specified in [MS-NSPI], or the Exchange Server NSPI Protocol, as specified in [MS-OXNSPI],<1> as the underlying transport protocol when handling address books online by using an NSPI server, and it uses the OAB Format and Schema Protocol, as specified in [MS-OXOAB], as the underlying structure when handling address books by using a local offline address book (OAB).

## 2.2 Message Syntax

When messaging clients are using an NSPI server, the properties described in this specification are returned and given to various NSPI functions of either the Name Service Provider Interface (NSPI) Protocol, as specified in [MS-NSPI] or the Exchange Server NSPI Protocol, as specified in [MS-OXNSPI]. The use of NSPI calls depends on the object type and its associated property list. As a collection, the entire set of NSPI calls enables messaging clients to access and browse address lists, and to manipulate or obtain data on Address Book objects in that list.

Alternatively, these structures are maintained in an offline address book (OAB) by using the OAB Format and Schema Protocol, as specified in [MS-OXOAB]. The OAB is a collection of address lists, each of which contains Address Book objects and their properties. Messaging clients use the OAB Format and Schema Protocol, as specified in [MS-OXOAB], to access and browse the address lists and to obtain data on the Address Book objects in those lists.

There are many types of Address Book objects, including but not limited to **mail users**, **distribution lists**, address book containers, **Resource objects**, **Department objects**, **Organization objects**, and templates. This protocol does not require values for all properties that are defined for any object type, nor does it limit the properties to those of their own type or those listed in this specification. When a data source, such as an OAB or an NSPI server, includes properties that are not in this set, the format and meaning of the property is defined by the implementation of that data source.

In all the properties of type **PtypString** ([MS-OXCDATA] section 2.11.1) that are listed in this document, a request to an NSPI server of that property with type **PtypString8** ([MS-OXCDATA] section 2.11.1) or **PtypString** is permitted. If a conversion is required, the string will be converted to the type requested by the client, as specified in either [MS-NSPI] or [MS-OXNSPI].

For all of the properties of type **PtypString** that are listed in this document, an OAB that includes a value for that string property MUST contain exactly one string representation. The internal representation of strings in an OAB that use the OAB Format and Schema Protocol is specified in [MS-OXOAB]. A client that looks up the value for a string property MUST convert the value to the string type that is native to the client before it interprets the value of that property.

## 2.2.1   Definitions

The following sections describe structures that are used by various Address Book object properties.

### 2.2.1.1   Distinguished Names for Objects

**Note**   *Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

Address book distinguished names (ABDNs) are used to uniquely identify objects in the address book. Throughout the rest of this specification, the term **distinguished name (DN) (1)** is used to refer to an address book DN (1). Each Address Book object MUST have a unique DN (1) value, expressed as a NULL-terminated **ASCII** string. The DN (1) is stored in the **PidTagEmailAddress** property (section 2.2.3.14). The DN (1) is also embedded in the **Distinguished Name** field of the PermanentEntryID structure, as specified in [MS-NSPI] and [MS-OXNSPI] section 2.3.8.3.<2> DNs (1) are structured as shown in the following **Augmented Backus-Naur Form (ABNF)** definition, as specified in [RFC5234].

```
dn = organization-dn /
  addresslist-dn /
  x500-dn
organization-dn= org-rdn
addresslist-dn  = "/guid=" container-guid /
  gal-addrlist-dn
container-guid= 32(HEXDIG)
gal-addrlist-dn = "/"
x500-dn    = x500-container-dn object-rdn
 ; x500-dns are limited to 16 levels
x500-container-dn = org-rdn org-unit-rdn 0*13(container-rdn)
org-rdn    = "/o=" rdn
org-unit-rdn = "/ou=" rdn
container-rdn = "/cn=" rdn
object-rdn  = "/cn=" rdn
rdn   = ( non-space-teletex ) /
                ( non-space-teletex *62(teletex-char)
  non-space-teletex )
 ; rdn values are limited to 64 characters
 ; the number of rdns is limited to 16 but the
 ; total cumulative length of rdn characters in
 ; An x500-dn is limited to 256.
teletex-char = SP / non-space-teletex
non-space-teletex= "!" / DQUOTE / "%" / "&" / "\" / "(" / ")" /
  "*" / "+" / "," / "-" / "." / "0" / "1" /
  "2" / "3" / "4" / "5" / "6" / "7" / "8" /
  "9" / ":" / "<" / "=" / ">" / "?" / "@" /
  "A" / "B" / "C" / "D" / "E" / "F" / "G" /
  "H" / "I" / "J" / "K" / "L" / "M" / "N" /
  "O" / "P" / "Q" / "R" / "S" / "T" / "U" /
  "V" / "W" / "X" / "Y" / "Z" / "[" / "]" /
  "_" / "a" / "b" / "c" / "d" / "e" / "f" /
  "g" / "h" / "i" / "j" / "k" / "l" / "m" /
  "n" / "o" / "p" / "q" / "r" / "s" / "t" /
  "u" / "v" / "w" / "x" / "y" / "z" / "|"
```

DNs (1) for specific objects have a strict format, as shown in the following table.

| Object type | DN format | Notes |
|---|---|---|
| Address book container | *addresslist-dn* | |
| **Global Address List** container | *gal-addrlist-dn* | |
| Mail user | *x500-dn* | The *org-rdn* string is the mail user's organization. |
| Organization | *organization-dn* | |
| Store | *x500-dn* | The *x500-container-dn* is the **mailbox** server. |
| Mailbox server | *x500-dn* | The **relative distinguished name (RDN) (2)** in the *object-rdn* is the name of the mailbox server. |
| Room container reference | *x500-dn* with no *container-rdn* | The RDN (2) of the *object-rdn* matches the *container-guid* of the address book container. |
| All other Address Book objects | *dn* | |

When the DN (1) of an Address Book object that is obtained from an NSPI server matches the DN (1) of an Address Book object that is obtained from an OAB, the objects represent the same entity. For such an object to be visible on both data sources, the properties available in the OAB SHOULD be a subset of the properties in the NSPI server, MAY<3> include additional properties, and SHOULD have the same value when present on both data sources. One exception to having the same value on both data sources is if the properties are truncated in the OAB, according to the limitations specified in [MS-OXOAB]. Another exception is if the value on an NSPI server has changed since the OAB was created, or if the NSPI server was restored from a backup after the OAB was created. In such a case, the NSPI server and the OAB are said to be "out of sync". That is, the data in the two sources reflects two different time periods.

## 2.2.2   Properties that Apply to Containers in the Address Book Hierarchy Table

*Note*   *Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

An address book is a collection of Address Book objects, each of which is contained in any number of address lists. The address book is arranged as a hierarchy of address book containers. Each address book container, in turn, describes an address list that contains many Address Book objects.

When using an offline address book (OAB), messaging clients SHOULD obtain information about the address book hierarchy and its address book containers by using the Offline Address Book Retrieval File Format, as specified in [MS-OXWOAB], or MAY<4> obtain information by using the Offline Address Book Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB]. Because an OAB maintains its own structure for the **hierarchy table**, none of the properties of address book containers described in this document apply to the OAB.

The hierarchy table is a set of rows, each of which describes one address book container. For details about retrieving the address book hierarchy table, see section 3.1.4.1.

## 2.2.2.1   PidTagContainerFlags

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagContainerFlags** property ([MS-OXPROPS] section 2.714) contains a bitmask of **flags** that describe capabilities of an address book container.

The flags listed in the following table are defined for the bitmask of the **PidTagContainerFlags** property. The **PidTagContainerFlags** property MUST NOT contain any other flags.

| Flag name | Value | Description |
|---|---|---|
| **AB_RECIPIENTS** | 0x00000001 | The container holds Address Book objects. This flag does not indicate whether any objects are actually present in the container. This flag MUST be set for all containers and distribution lists that are returned by the NSPI server. |
| **AB_SUBCONTAINERS** | 0x00000002 | The container holds child containers. This flag does not indicate whether any subcontainers are actually present in the container. |
| **AB_UNMODIFIABLE** | 0x00000008 | It is not possible to add or remove Address Book objects from the container. This flag MUST be set for containers that are returned by the NSPI server. |

For distribution lists that are obtained from an NSPI server, this property MUST be present, and its value MUST be AB_RECIPIENTS.

Because this property applies to a container in the hierarchy table, it is not present on objects in the offline address book (OAB). An OAB has its own structure for maintaining the hierarchy, using the OAB Retrieval File Format, as specified in [MS-OXWOAB], and the OAB Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB].

### 2.2.2.2  PidTagDepth

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagDepth** property ([MS-OXCTABL] section 2.2.1.4) represents the relative level of depth of a container in a hierarchy table. Objects in the hierarchy table that share the same value for the **PidTagDepth** property SHOULD be considered peer containers by clients if they are adjacent or if they are separated only by containers that have equivalent or larger depth values.

Because this property applies to a container in the hierarchy table, it is not present on objects in an offline address book (OAB). An OAB has its own structure for maintaining the hierarchy, using the OAB Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB].

### 2.2.2.3  PidTagAddressBookContainerId

*Note   Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookContainerId** property ([MS-OXPROPS] section 2.582) identifies a container on an NSPI server. Note that this property is a **Minimal Entry ID**. A value of 0 (zero) represents the Global Address List (GAL). This value is used in other NSPI calls (such as the **NspiResolveNamesW** method call) to identify which container the NSPI call applies. If the value is nonzero, it is only a valid representation of the specific container. This representation lasts for the time that the connection to the NSPI server lasts, or, after disconnection from and reconnection to the same or another NSPI server that is using the **NspiBind** property, for as long as the new server

identifies itself as having the same **GUID** in its return value for the *pServerGuid* parameter, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.1.<5>

Because this property applies to a container in the hierarchy table, it is not present on objects in an offline address book (OAB). An OAB has its own structure for maintaining the hierarchy, using the OAB Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB].

### 2.2.2.4   PidTagAddressBookIsMaster

Data type: **PtypBoolean** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookIsMaster** property ([MS-OXPROPS] section 2.615) is **TRUE** (0x01) if it is possible to create Address Book objects in that container, and **FALSE** (0x00) otherwise. The value does not pertain to parent containers or subcontainers of this container.

Because this property applies to a container in the hierarchy table, it is not present on objects in an offline address book (OAB). An OAB has its own structure for maintaining the hierarchy, using the OAB Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB].

### 2.2.2.5   PidTagAddressBookParentEntryId

Data type: **PtypBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookParentEntryId** property ([MS-OXPROPS] section 2.629) is the **entry ID** of the parent container in a hierarchy of address book containers. This property is not present if no parent container exists. Messaging clients use this property to expand and collapse a hierarchy of address book containers in an address book hierarchy table.

Because this property applies to a container in the hierarchy table, it is not present on objects in an offline address book (OAB). An OAB has its own structure for maintaining the hierarchy, using the OAB Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB].

### 2.2.3   Properties that Apply to All Address Book Objects

*Note   Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

Address Book objects are listed in an offline address book (OAB) by the OAB Format and Schema Protocol, as specified in [MS-OXOAB], or accessed on an NSPI server by using various Name Service Provider Interface (NSPI) Protocol method calls, as specified in [MS-NSPI] and [MS-OXNSPI].<6> Each object represents any addressable entity, including but not limited to mail user, distribution list, Department object, Organization object, address book container, or Resource object.

It is possible to set any Address Book object as an addressee of an outbound message to be sent by a messaging client. A mail user is an example, and it can contain an e-mail address to be used for messaging. A distribution list is a special type of Address Book object that represents a collection of other Address Book objects, can be an addressee of an outbound message, and SHOULD have additional properties and operations, as specified in section 2.2.6. Mail user objects are further broken down into subtypes, such as rooms, equipment, messaging forums, or other types. Note that servers can restrict client access to specific properties as required by the implementation. One example is distribution list membership; the server can restrict access to distribution lists in which the members are hidden.

The Department object and the Organization object are types of Address Book objects that are required to support a hierarchical address book view that messaging clients need.

### 2.2.3.1 PidTagDisplayName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagDisplayName** property ([MS-OXCFOLD] section 2.2.2.2.2.4) represents a displayable form of an Address Book object. When the Address Book object is a mail user, the **PidTagDisplayName** property is the full name of the mail user. When the object is a distribution list, the value of the **PidTagDisplayName** property is the name of the distribution list that would be displayed when addressing messages to that distribution list. When the object is a Department object, the **PidTagDisplayName** property value contains the name of that department. When the object is any other Address Book object, the value of the **PidTagDisplayName** property is the displayable name of that object.

The **PidTagDisplayName** property is also one of the columns that are returned for the set of address book containers in the address book hierarchy table. The value of the **PidTagDisplayName** property for each row in the hierarchy table represents the name of the address book container of that row.

### 2.2.3.2 PidTagEntryId

*Note   Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

Data type: **PtypBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagEntryId** property ([MS-OXCPERM] section 2.2.4) is used to identify many different types of **messaging objects**, including objects that are not in an address book. The formats of the **PidTagEntryId** property for nonaddressing messaging objects are specified in [MS-OXCDATA]. These other formats do not apply to any objects in an offline address book (OAB) or on an NSPI server.

The **PidTagEntryId** property identifies an Address Book object on an NSPI server and represents either the PermanentEntryID structure, as specified in [MS-NSPI] and [MS-OXNSPI] section 2.3.8.3, or the **EphemeralEntryID** structure, as specified in [MS-NSPI] and [MS-OXNSPI] section 2.3.8.2.<7> Messaging clients use this property to open an object and to perform operations on it, such as obtaining other properties. The types of operations that can be performed on an object are specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4. When the object is in **Permanent Entry ID** format, its DN (1) MUST match the value for the **PidTagEmailAddress** property (section 2.2.3.14) and MUST follow the format that is particular to the type of object, as described in section 2.2.1.1.

The OAB Format and Schema Protocol specification, as specified in [MS-OXOAB], does not include value for the **PidTagEntryId** property for Address Book objects in its data structure. Instead, the **PidTagEmailAddress** property (section 2.2.3.14) identifies objects in an OAB.

### 2.2.3.3 PidTagTemplateid

Data type: **PtypBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagTemplateid** property ([MS-OXPROPS] section 2.1104) contains the **PidTagEntryId** property (section 2.2.3.2), expressed in Permanent Entry ID format. This value MUST be present for all Address Book objects on an NSPI server, its DN (1) MUST match the value for the **PidTagEmailAddress** property (section 2.2.3.14), and its DN (1) MUST follow the format particular to the type of object, as described in section 2.2.1.1.

The **PidTagTemplateid** property is not present on objects in an offline address book (OAB).

### 2.2.3.4 PidTagRecordKey

Data type: **PtypBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagRecordKey** property ([MS-OXCPRPT] section 2.2.1.8) contains a unique binary-comparable identifier for an Address Book object. This value MUST be present on all objects on an NSPI server and MUST match the value for the **PidTagTemplateid** property (section 2.2.3.3).

The **PidTagRecordKey** property is not present on objects in an offline address book (OAB).

### 2.2.3.5 PidTagSearchKey

Data type: **PtypBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagSearchKey** property ([MS-OXCPRPT] section 2.2.1.9) is formed by concatenating the ASCII string "EX: " followed by the DN (1) for the object converted to all uppercase, followed by a zero-byte value. This value MUST be present for all Address Book objects on an NSPI server and MUST be in the aforementioned format.

The **PidTagSearchKey** property is not present on objects in an offline address book (OAB).

### 2.2.3.6 PidTagInstanceKey

Data type: **PtypBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagInstanceKey** property ([MS-OXPROPS] section 2.810) identifies an object on an NSPI server. It is a Minimal Entry ID, represented as a 4 byte binary value, in little-endian byte order.

The **PidTagInstanceKey** property is not present on objects in an offline address book (OAB).

### 2.2.3.7 PidTagAddressBookDisplayNamePrintable

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookDisplayNamePrintable** property ([MS-OXCMSG] section 2.2.1.30) contains a displayable form of an Address Book object that can be rendered in the client user's own **code page**.

### 2.2.3.8 PidTagTransmittableDisplayName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagTransmittableDisplayName** property ([MS-OXPROPS] section 2.1111) contains an Address Book object's display name that is transmitted with the message. This value MUST be present on all objects on an NSPI server, and its value MUST match the value for the **PidTagDisplayName** property (section 2.2.3.1).

The **PidTagTransmittableDisplayName** property is not present on objects in an offline address book (OAB).

### 2.2.3.9 PidTagAddressBookPhoneticDisplayName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookPhoneticDisplayName** property ([MS-OXPROPS] section 2.632) contains the phonetic representation of the **PidTagDisplayName** property ([MS-OXCFOLD] section 2.2.2.2.2.4).<8>

### 2.2.3.10 PidTagObjectType

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagObjectType** property ([MS-OXCPRPT] section 2.2.1.7) contains a value that specifies the type of an object.

The **PidTagObjectType** property MUST be present for all Address Book objects and MUST have one of the values listed in the following table.

| Value name | Value | Description |
| --- | --- | --- |
| **MAILUSER** | 0x00000006 | A mail user, or any Address Book object that is not a distribution list or forum. |
| **DISTLIST** | 0x00000008 | A distribution list. |
| **folder** | 0x00000003 | A messaging forum, such as a bulletin board service or a public or shared folder. |

### 2.2.3.11 PidTagDisplayType

*Note   Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagDisplayType** property ([MS-OXPROPS] section 2.748) contains a value that indicates how to display an Address Book object in a table or as an addressee on a message. Messaging clients use this property to display an icon, make the object bold, or apply some other display element to make it easy for a user who is viewing the object to distinguish its type. In addition, the **PidTagDisplayTypeEx** property (section 2.2.3.12) provides a means to refine the display further.

The **PidTagDisplayType** property MUST have one of the following display type values, as specified in [MS-NSPI] and [MS-OXNSPI] section 2.2.3, according to the object's type: **DT_MAILUSER**, **DT_DISTLIST**, **DT_FORUM**, **DT_AGENT**, **DT_ORGANIZATION**, **DT_PRIVATE_DISTLIST**, or **DT_REMOTE_MAILUSER**.<9>

### 2.2.3.12 PidTagDisplayTypeEx

*Note   Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagDisplayTypeEx** property ([MS-OXPROPS] section 2.749) contains a value that indicates how to display an Address Book object in a table or as a **recipient (2)** on a message. Messaging clients use this to display an icon, make the object bold, or apply some other display element to make it easy for a user who is viewing the object to distinguish its type. This property contains more

detailed information about the object's display information than does the **PidTagDisplayType** property (section 2.2.3.11). In addition to the display information contained in the **PidTagDisplayType** property, this property distinguishes among additional object types. When the object comes from a remote server, the **PidTagDisplayType** property also includes information about the type of object on that remote server, as well as the type on the local server.

The **PidTagDisplayTypeEx** property also includes information that indicates whether it is possible to share information from the user's own mailbox to the entity represented by an Address Book object.

The **PidTagDisplayTypeEx** property is a bitmask of flags and values and has the following structure:

**r (1 bit)**: 1 indicates that the value in **dtRemote** is the remote display type. The numeral 0 (zero) means that **dtRemote** is undefined. This represents the high order bit.

**s (1 bit)**: 1 indicates that the mailbox server supports sharing to the entity that an Address Book object represents. The numeral 0 (zero) means that it does not support such sharing.

**reserved (14 bits)**: Undefined. This value MUST contain all zeros and MUST be ignored by clients.

**dtRemote (1 byte)**: The display type of an Address Book object in the remote forest. This is undefined if **r** is 0 (zero). If **r** is 1, it contains one of the values listed in the following table.

**dtLocal (1 byte)**: The display type of an Address Book object in the messaging user's local forest. It contains one of the values listed in the following table.

Note that **dtLocal** and/or **dtRemote** MUST have one of the following display type values, as defined in [MS-NSPI] and [MS-OXNSPI] section 2.2.3,<10> according to the object's type: **DT_MAILUSER**, **DT_DISTLIST**, **DT_FORUM**, **DT_AGENT**, **DT_ORGANIZATION**, **DT_PRIVATE_DISTLIST**, **DT_REMOTE_MAILUSER**, or one of the values listed in the following table.

| Value name | Value | Description |
|---|---|---|
| **DT_ROOM** | 0x07 | A conference room. It is possible for messaging clients to send **meeting requests** to the specified Address Book object to book the room. |
| **DT_EQUIPMENT** | 0x08 | Equipment. It is possible for messaging clients to send meeting requests to the specified Address Book object to reserve the equipment. |
| **DT_SEC_DISTLIST** | 0x09 | A distribution list used as a security group on the server. |

### 2.2.3.13   PidTagAddressType

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressType** property ([MS-OXPROPS] section 2.646) contains an Address Book object's e-mail **address type**. It MUST have the value "EX" for all objects on an NSPI server.

The **PidTagAddressType** property is not present on objects in an offline address book (OAB).

### 2.2.3.14   PidTagEmailAddress

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagEmailAddress** property ([MS-OXPROPS] section 2.750) contains an Address Book object's e-mail address, expressed in X500 format, using the format that is particular to the type of

object, as specified in section 2.2.1.1. This property MUST be present for every Address Book object. Its value MUST match the DN (1) of the Permanent Entry ID for the object if the object is present on an NSPI server. Its DN (1) MUST follow the format particular to the type of object, as specified in section 2.2.1.1.

### 2.2.3.15 PidTagAddressBookObjectDistinguishedName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookObjectDistinguishedName** property ([MS-OXPROPS] section 2.624) contains the DN (1) of an Address Book object in DN (1) format, as specified in section 2.2.1.1. If present, its DN (1) MUST follow the format that is particular to the type of object, as specified in section 2.2.1.1. Its value, if present, MUST match the value for the **PidTagEmailAddress** property. This value MUST be present on all Address Book objects on an NSPI server.

### 2.2.3.16 PidTagCreationTime

Data type: **PtypTime** ([MS-OXCDATA] section 2.11.1)

The **PidTagCreationTime** property ([MS-OXCMSG] section 2.2.2.3) contains the creation date and time for an Address Book object in **Coordinated Universal Time (UTC)**.

### 2.2.3.17 PidTagLastModificationTime

Data type: **PtypTime** ([MS-OXCDATA] section 2.11.1)

The **PidTagLastModificationTime** property ([MS-OXCMSG] section 2.2.2.2) contains the date and time that an Address Book object was last modified in Coordinated Universal Time (UTC).

### 2.2.3.18 PidTagSendRichInfo

Data type: **PtypBoolean** ([MS-OXCDATA] section 2.11.1)

The **PidTagSendRichInfo** property ([MS-OXPROPS] section 2.1072) contains **TRUE** (0x01) if the e-mail-enabled entity represented by an Address Book object can receive all message content, including **Rich Text Format (RTF)** and other embedded objects. When sending mail by using the [RFC2822] Internet message format and **Multipurpose Internet Mail Extensions (MIME)** to the E-mail Object Conversion Protocol, as specified in [MS-OXCMAIL], the **PidTagSendRichInfo** property specifies whether to encode the message in MIME or in **Transport Neutral Encapsulation Format (TNEF)**, as specified in [MS-OXCMAIL]. The **PidTagSendRichInfo** property contains **FALSE** (0x00) if the e-mail-enabled entity cannot receive formatted message content.

### 2.2.3.19 PidTagSendInternetEncoding

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagSendInternetEncoding** property ([MS-OXPROPS] section 2.1071) contains a bitmask of message-encoding preferences for mail sent to an e-mail-enabled entity that is represented by an Address Book object. When sending mail by using the [RFC2822] Internet message format and Multipurpose Internet Mail Extensions (MIME) to the E-mail Object Conversion Protocol, as specified in [MS-OXCMAIL], the **PidTagSendInternetEncoding** property (section 2.2.3.19) specifies the format of the MIME body, as specified in [MS-OXCMAIL].

### 2.2.3.20   PidTagAccount

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAccount** property ([MS-OXOCNTC] section 2.2.1.10.11) contains an Address Book object's **alias**, which is an alternative name by which the object can be identified.

### 2.2.3.21   PidTagSmtpAddress

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagSmtpAddress** property ([MS-OXPROPS] section 2.1081) contains an Address Book object's **Simple Mail Transfer Protocol (SMTP)** address.

### 2.2.3.22   PidTagAddressBookTargetAddress

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookTargetAddress** property ([MS-OXPROPS] section 2.643) contains the foreign system e-mail address of an Address Book object. If this value is present, it MUST be prefixed with the address type of the foreign address, followed by a colon (":"), followed by the foreign e-mail address expressed in that address type. For Simple Mail Transfer Protocol (SMTP) addresses, this value is "SMTP:" followed by the foreign SMTP address.

### 2.2.3.23   PidTagAddressBookProxyAddresses

Data type: **PtypMultipleString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookProxyAddresses** property ([MS-OXPROPS] section 2.635) contains alternate e-mail addresses for an Address Book object. Each string MUST be prefixed by an e-mail type, followed by a colon (":"), followed by the e-mail address in a format for that e-mail type. For a Simple Mail Transfer Protocol (SMTP) address, the string MUST start with the string "SMTP:" or "SMTP:" followed by an SMTP address. The first string starting with the uppercase string "SMTP:" MUST match the **PidTagSmtpAddress** property (section 2.2.3.21) for the Address Book object.

### 2.2.3.24   PidTagAddressBookSeniorityIndex

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookSeniorityIndex** property ([MS-OXPROPS] section 2.642) contains a signed integer that specifies the seniority order of Address Book objects that represent members of a department and are referenced by a Department object or **departmental group**, with larger values specifying members that are more senior. It also specifies the sort order of Department objects or departmental groups that represent child departments of another parent department in a hierarchy of departments and that are referenced by the parent's Department object or departmental group, in descending order.

### 2.2.3.25   PidTagAddressBookObjectGuid

Data type: **PtypBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookObjectGuid** property ([MS-OXPROPS] section 2.625) contains a GUID that uniquely identifies an Address Book object.

### 2.2.3.26 PidTagAddressBookSenderHintTranslations

Data type: **PtypMultipleString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookSenderHintTranslations** property<11> ([MS-OXPROPS] section 2.641) contains the **locale** ID and translations of the default **mail tip**. An example of this value is "en-US:Hello" or "es:Hola".

### 2.2.3.27 PidTagAddressBookDeliveryContentLength

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookDeliveryContentLength** property<12> ([MS-OXPROPS] section 2.583) specifies the maximum size of a message that a recipient (2) can receive.

### 2.2.3.28 PidTagAddressBookModerationEnabled

Data type: **PtypBoolean** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookModerationEnabled** property<13> ([MS-OXPROPS] section 2.622) contains **TRUE** (0x01) if moderation is enabled for the mail user or distribution list; otherwise, this property contains **FALSE** (0x00).

### 2.2.3.29 PidTagAddressBookDistributionListMemberCount

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookDistributionListMemberCount** property<14> ([MS-OXPROPS] section 2.587) contains the total number of recipients (2) in a specified distribution list. This value includes the members of all the distribution lists that are members of the specified distribution list.

### 2.2.3.30 PidTagAddressBookDistributionListExternalMemberCount

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookDistributionListExternalMemberCount** property<15> ([MS-OXPROPS] section 2.586) contains the number of external recipients (2) in a specified distribution list.

### 2.2.3.31 PidTagComment

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagComment** property ([MS-OXCFOLD] section 2.2.2.2.2.2) contains a comment about the purpose or content of an Address Book object.

### 2.2.3.32 PidTagMappingSignature

Data type: **PtypBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagMappingSignature** property ([MS-OXPROPS] section 2.847) MUST be present on all Address Book objects on an NSPI server and MUST contain the following 16 bytes:

```
0xDC, 0xA7, 0x40, 0xC8, 0xC0, 0x42, 0x10, 0x1A, 0xB4, 0xB9, 0x08, 0x00, 0x2B, 0x2F, 0xE1,
0x82
```

The **PidTagMappingSignature** property is not present on objects in an offline address book (OAB).

### 2.2.3.33   PidTagInitialDetailsPane

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagInitialDetailsPane** property ([MS-OXPROPS] section 2.807) indicates which page of a **display template** to display first. This value MUST be present on all Address Book objects on an NSPI server, and it MUST have the value 0 (zero).

### 2.2.3.34   PidTagAddressBookExtensionAttribute1 through PidTagAddressBookExtensionAttribute15

These 15 properties are used to store custom values. These properties are **PtypString** values ([MS-OXCDATA] section 2.11.1) that are defined and populated by the organization that modified the display templates.

### 2.2.3.35   PidTagAddressBookDisplayTypeExtended

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookDisplayTypeExtended** property ([MS-OXPROPS] section 2.585) is limited to **Recipient objects**. The **PidTagAddressBookDisplayTypeExtended** property SHOULD be present on objects on an NSPI server or an offline address book (OAB).<16>

## 2.2.4   Properties that Apply to Mail User Objects

### 2.2.4.1   PidTagSurname

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagSurname** property ([MS-OXPROPS] section 2.1097) contains a mail user's family name.

### 2.2.4.2   PidTagGivenName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagGivenName** property ([MS-OXPROPS] section 2.782) contains a mail user's given name.

### 2.2.4.3   PidTagNickname

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagNickname** property ([MS-OXPROPS] section 2.874) contains a mail user's nickname.

### 2.2.4.4   PidTagDisplayNamePrefix

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagDisplayNamePrefix** property ([MS-OXPROPS] section 2.746) contains a mail user's honorific title.

### 2.2.4.5   PidTagInitials

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagInitials** property ([MS-OXPROPS] section 2.808) contains the initials for parts of a mail user's full name.

### 2.2.4.6   PidTagGeneration

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagGeneration** property ([MS-OXPROPS] section 2.781) contains a generational abbreviation that follows a mail user's full name.

### 2.2.4.7   PidTagTitle

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagTitle** property ([MS-OXPROPS] section 2.1108) contains a mail user's job title.

### 2.2.4.8   PidTagOfficeLocation

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagOfficeLocation** property ([MS-OXPROPS] section 2.879) contains a mail user's office location.

### 2.2.4.9   PidTagDepartmentName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagDepartmentName** property ([MS-OXPROPS] section 2.741) contains a name for the department in which a mail user works.

### 2.2.4.10   PidTagCompanyName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagCompanyName** property ([MS-OXPROPS] section 2.709) contains a mail user's company name.

### 2.2.4.11   PidTagAssistant

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAssistant** property ([MS-OXPROPS] section 2.652) contains the name of a mail user's administrative assistant.

### 2.2.4.12   PidTagManagerName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagManagerName** property ([MS-OXPROPS] section 2.846) contains the name of a mail user's manager.

### 2.2.4.13 PidTagAddressBookManagerDistinguishedName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookManagerDistinguishedName** property ([MS-OXPROPS] section 2.619) contains the DN (1) of a mail user's manager.

### 2.2.4.14 PidTagAddressBookPhoneticGivenName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookPhoneticGivenName** property ([MS-OXPROPS] section 2.633) contains the phonetic representation of the **PidTagGivenName** property (section 2.2.4.2).<17>

### 2.2.4.15 PidTagAddressBookPhoneticSurname

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookPhoneticSurname** property ([MS-OXPROPS] section 2.634) contains the phonetic representation of the **PidTagSurname** property (section 2.2.4.1).<18>

### 2.2.4.16 PidTagAddressBookPhoneticCompanyName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookPhoneticCompanyName** property ([MS-OXPROPS] section 2.630) contains the phonetic representation of the **PidTagCompanyName** property (section 2.2.4.10).<19>

### 2.2.4.17 PidTagAddressBookPhoneticDepartmentName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookPhoneticDepartmentName** property ([MS-OXPROPS] section 2.631) contains the phonetic representation of the **PidTagDepartmentName** property (section 2.2.4.9).<20>

### 2.2.4.18 PidTagPostalAddress

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagPostalAddress** property ([MS-OXPROPS] section 2.929) contains a mail user's postal address.

### 2.2.4.19 PidTagStreetAddress

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagStreetAddress** property ([MS-OXPROPS] section 2.1092) contains a mail user's street address.

### 2.2.4.20 PidTagPostOfficeBox

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagPostOfficeBox** property ([MS-OXPROPS] section 2.931) contains the number or identifier of a mail user's post office box.

### 2.2.4.21   PidTagLocality

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagLocality** property ([MS-OXPROPS] section 2.842) contains the name of a mail user's locality, such as the town or city.

### 2.2.4.22   PidTagStateOrProvince

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagStateOrProvince** property ([MS-OXPROPS] section 2.1088) contains the name of a mail user's state or province.

### 2.2.4.23   PidTagPostalCode

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagPostalCode** property ([MS-OXPROPS] section 2.930) contains the postal code for a mail user's postal address.

### 2.2.4.24   PidTagCountry

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagCountry** property ([MS-OXPROPS] section 2.723) contains the name of a mail user's country/region.

### 2.2.4.25   PidTagHomeAddressStreet

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagHomeAddressStreet** property ([MS-OXPROPS] section 2.797) contains a mail user's home street address.

### 2.2.4.26   PidTagHomeAddressPostOfficeBox

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagHomeAddressPostOfficeBox** property ([MS-OXPROPS] section 2.795) contains the number or identifier of a mail user's home post office box.

### 2.2.4.27   PidTagHomeAddressCity

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagHomeAddressCity** property ([MS-OXPROPS] section 2.792) contains the name of a mail user's home locality, such as the town or city.

### 2.2.4.28   PidTagHomeAddressStateOrProvince

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagHomeAddressStateOrProvince** property ([MS-OXPROPS] section 2.796) contains the name of a mail user's home state or province.

### 2.2.4.29 PidTagHomeAddressPostalCode

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagHomeAddressPostalCode** property ([MS-OXPROPS] section 2.794) contains the postal code for a mail user's home postal address.

### 2.2.4.30 PidTagHomeAddressCountry

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagHomeAddressCountry** property ([MS-OXPROPS] section 2.793) contains the name of a mail user's home country/region.

### 2.2.4.31 PidTagOtherAddressStreet

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagOtherAddressStreet** property ([MS-OXPROPS] section 2.918) contains a mail user's other street address.

### 2.2.4.32 PidTagOtherAddressPostOfficeBox

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagOtherAddressPostOfficeBox** property ([MS-OXPROPS] section 2.916) contains the number or identifier of a mail user's other post office box.

### 2.2.4.33 PidTagOtherAddressCity

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagOtherAddressCity** property ([MS-OXPROPS] section 2.913) contains the name of a mail user's other locality, such as the town or city.

### 2.2.4.34 PidTagOtherAddressStateOrProvince

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagOtherAddressStateOrProvince** property ([MS-OXPROPS] section 2.917) contains the name of a mail user's other state or province.

### 2.2.4.35 PidTagOtherAddressPostalCode

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagOtherAddressPostalCode** property ([MS-OXPROPS] section 2.915) contains the postal code for a mail user's other postal address.

### 2.2.4.36 PidTagOtherAddressCountry

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagOtherAddressCountry** property ([MS-OXPROPS] section 2.914) contains the name of a mail user's other country/region.

### 2.2.4.37   PidTagPrimaryTelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagPrimaryTelephoneNumber** property ([MS-OXPROPS] section 2.935) contains a mail user's primary telephone number.

### 2.2.4.38   PidTagBusinessTelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagBusinessTelephoneNumber** property ([MS-OXPROPS] section 2.696) contains the primary telephone number of a mail user's place of business.

### 2.2.4.39   PidTagHomeTelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagHomeTelephoneNumber** property ([MS-OXPROPS] section 2.799) contains the primary telephone number of a mail user's home.

### 2.2.4.40   PidTagBusiness2TelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagBusiness2TelephoneNumber** property ([MS-OXPROPS] section 2.692) contains a secondary telephone number at a mail user's place of business.

### 2.2.4.41   PidTagBusiness2TelephoneNumbers

Data type: **PtypMultipleString** ([MS-OXCDATA] section 2.11.1)

The **PidTagBusiness2TelephoneNumbers** property ([MS-OXPROPS] section 2.693) contains secondary telephone numbers at a mail user's place of business.

### 2.2.4.42   PidTagHome2TelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagHome2TelephoneNumber** property ([MS-OXPROPS] section 2.790) contains a secondary telephone number at a mail user's home.

### 2.2.4.43   PidTagHome2TelephoneNumbers

Data type: **PtypMultipleString** ([MS-OXCDATA] section 2.11.1)

The **PidTagHome2TelephoneNumbers** property ([MS-OXPROPS] section 2.791) contains secondary telephone numbers at a mail user's home.

### 2.2.4.44   PidTagCallbackTelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagCallbackTelephoneNumber** property ([MS-OXPROPS] section 2.697) contains a callback telephone number for a mail user.

### 2.2.4.45   PidTagMobileTelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagMobileTelephoneNumber** property ([MS-OXPROPS] section 2.871) contains a mail user's mobile telephone number.

### 2.2.4.46   PidTagRadioTelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagRadioTelephoneNumber** property ([MS-OXPROPS] section 2.945) contains a mail user's radio telephone number.

### 2.2.4.47   PidTagCarTelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagCarTelephoneNumber** property ([MS-OXPROPS] section 2.699) contains a mail user's car telephone number.

### 2.2.4.48   PidTagOtherTelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagOtherTelephoneNumber** property ([MS-OXPROPS] section 2.919) contains a mail user's alternate telephone number.

### 2.2.4.49   PidTagPagerTelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagPagerTelephoneNumber** property ([MS-OXPROPS] section 2.922) contains a mail user's pager telephone number.

### 2.2.4.50   PidTagPrimaryFaxNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagPrimaryFaxNumber** property ([MS-OXPROPS] section 2.933) contains the telephone number of a mail user's primary fax machine.

### 2.2.4.51   PidTagBusinessFaxNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagBusinessFaxNumber** property ([MS-OXPROPS] section 2.694) contains the telephone number of a mail user's business fax machine.

### 2.2.4.52   PidTagHomeFaxNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagHomeFaxNumber** property ([MS-OXPROPS] section 2.798) contains the telephone number of a mail user's home fax machine.

### 2.2.4.53 PidTagCompanyMainTelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagCompanyMainTelephoneNumber** property ([MS-OXPROPS] section 2.708) contains the main telephone number of a mail user's company.

### 2.2.4.54 PidTagTelecommunicationsDeviceForDeafTelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagTelecommunicationsDeviceForDeafTelephoneNumber** property ([MS-OXPROPS] section 2.1101) contains a mail user's telecommunication device for the deaf (TTY/TDD) telephone number.

### 2.2.4.55 PidTagTelexNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagTelexNumber** property ([MS-OXPROPS] section 2.1102) contains a mail user's telex number.

### 2.2.4.56 PidTagIsdnNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagIsdnNumber** property ([MS-OXPROPS] section 2.824) contains the **Integrated Services Digital Network (ISDN)** telephone number for a mail user.

### 2.2.4.57 PidTagAssistantTelephoneNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAssistantTelephoneNumber** property ([MS-OXPROPS] section 2.653) contains the telephone number of a mail user's administrative assistant.

### 2.2.4.58 PidTagKeyword

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagKeyword** property ([MS-OXPROPS] section 2.830) contains a keyword that identifies a mail user to his or her system administrator.

### 2.2.4.59 PidTagGovernmentIdNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagGovernmentIdNumber** property ([MS-OXPROPS] section 2.783) contains a mail user's government identifier.

### 2.2.4.60 PidTagMessageHandlingSystemCommonName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagMessageHandlingSystemCommonName** property ([MS-OXPROPS] section 2.859) contains the **common name** of a messaging user for use in a message header. This property is read-only.

### 2.2.4.61 PidTagLanguage

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagLanguage** property ([MS-OXPROPS] section 2.831) contains a value that indicates the language in which a messaging user writes messages.

### 2.2.4.62 PidTagLocation

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagLocation** property ([MS-OXPROPS] section 2.843) contains the location of a mail user in a format that is useful to that person's organization.

### 2.2.4.63 PidTagOrganizationalIdNumber

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagOrganizationalIdNumber** property ([MS-OXPROPS] section 2.887) contains an identifier for a mail user that is used within that person's organization.

### 2.2.4.64 PidTagUserCertificate

Data type: **PtypBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagUserCertificate** property ([MS-OXPROPS] section 2.1116) has been deprecated. This property MUST be ignored by clients.

### 2.2.4.65 PidTagAddressBookX509Certificate

Data type: **PtypMultipleBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookX509Certificate** property ([MS-OXPROPS] section 2.645) contains ASN.1 DER encoded X.509 certificates for a mail user. Each binary value MUST be an ASN.1 DER encoded X.509 certificate, as specified in [RFC3280].

### 2.2.4.66 PidTagUserX509Certificate

Data type: **PtypMultipleBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagUserX509Certificate** property ([MS-OXPROPS] section 2.1118) contains a list certificate for a mail user. Each binary value MUST be either an ASN.1 DER encoded SignedData Type **binary large object (BLOB)** that contains the user's certificates and is signed with the user's certificate, as specified in [RFC3852], or a binary property as specified in the following paragraph.

To determine which of the preceding types each binary value is, the application MUST examine the first byte of each binary value. If the first byte has the value 0x30, it is an ASN.1 DER encoded SignedData Type BLOB. Otherwise, the binary value MUST be interpreted according to the format specified in this section. The client and the server SHOULD use the **PidTagUserX509Certificate** property instead of the **PidTagAddressBookX509Certificate** property (section 2.2.4.65) when looking for certificates.

**Non-ASN.1 Binary Value Format**

If the binary value is not an ASN.1 DER encoded SignedData Type BLOB, it MUST be a BLOB containing a set of security settings as specified in sections 2.2.4.66.1 through 2.2.4.66.12, one after another, in a continuous block of data. All settings in these sections MUST appear no more than once in the binary value unless stated otherwise. Each security setting has the format shown in the following packet diagram.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 1 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 2 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 3 0 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TAG | | | | | | | | | | | | | | | | LENGTH | | | | | | | | | | | | | | | |
| DATA (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
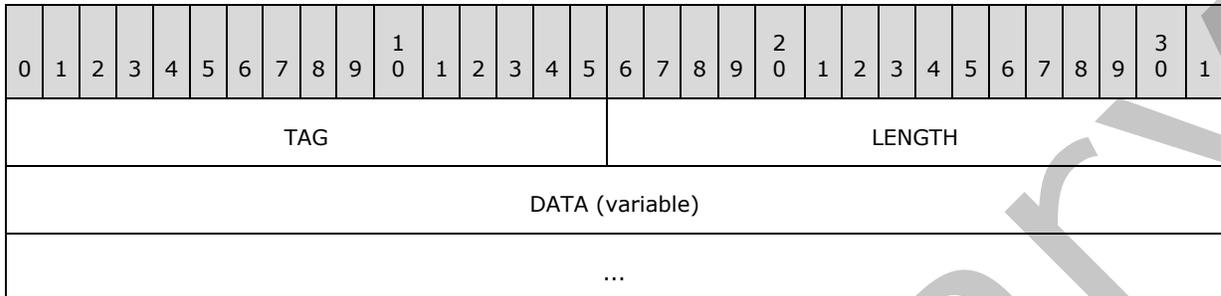
**TAG (2 bytes):** An unsigned value that identifies this particular security setting. If the **TAG** value found in a security setting is not among the ones listed in sections 2.2.4.66.1 through 2.2.4.66.12, the client and the server MUST ignore these settings.

**LENGTH (2 bytes):** The total length of the security setting, including the **TAG** field, the **LENGTH** field, and the **DATA** field.

**DATA (variable):** Contains the data associated with this security setting. Its length, in bytes, can be computed from the value of the **LENGTH** field.The following sections specify the security settings that appear in a non-ASN.1 certificate.

### 2.2.4.66.1   Property Version

The **TAG** value for this setting is 0x0001. The **LENGTH** value for this setting MUST be 0x0008. The **DATA** value specifies the version of the certificate. The unsigned **PtypInteger32 DATA** value MUST be 0x00000001. This setting MUST be part of each **PtypBinary** value.

### 2.2.4.66.2   Encryption Type

The **TAG** value for this setting is 0x0006. The **LENGTH** value for this setting MUST be 0x0008. The unsigned **PtypInteger32** ([MS-OXCDATA] section 2.11.1) **DATA** value specifies the type of encryption to be used with this certificate. A value of 0x00000001 specifies that the encryption type is **S/MIME (Secure/Multipurpose Internet Mail Extensions)**, as specified in [RFC3852]. A **DATA** value of 0x00000006 specifies that the encryption type is Fortezza, as specified in [RFC2876]. All other values do not have any defined meanings and MUST be ignored by both the server and the client. This setting MUST be part of each **PtypBinary** ([MS-OXCDATA] section 2.11.1) value.

### 2.2.4.66.3   Defaults

The **TAG** value for this setting is 0x0020. The **LENGTH** value for this setting MUST be 0x0008. The **DATA** field is a 4-byte bit field, which can contain either or both of the bit values listed in the following table.

| Bit | Meaning |
|---|---|
| 0x00000001 | This is the default certificate for S/MIME (Secure/Multipurpose Internet Mail Extensions). |
| 0x00000002 | This is the default certificate for all formats. If this bit is set, 0x00000001 MUST also be set. |

Any other bit flags set on this setting MUST be ignored. This setting MUST be part of each **PtypBinary** ([MS-OXCDATA] section 2.11.1) value.

### 2.2.4.66.4 ASCII Display Name

The **TAG** value for this setting is 0x000B. The **LENGTH** value for this setting MUST be larger than 4. The **DATA** field contains the user-readable name, as a NULL terminated ASCII string. Either the ASCII display name or the Unicode display name (as specified in section 2.2.4.66.5) — but not both — SHOULD be part of the binary value.<21>

### 2.2.4.66.5 Unicode Display Name

The **TAG** value for this setting is 0x0051. The **LENGTH** value for this setting MUST be larger than 4. The **DATA** field contains the user-readable name, as a NULL-terminated Unicode string. Either the Unicode display name or the ASCII display name — but not both — SHOULD be part of the certificate.<22>

### 2.2.4.66.6 KeyExSHA1Hash

The **TAG** value for this setting is 0x0022. The **LENGTH** value for this setting MUST be larger than 4. The **DATA** field contains the SHA1 Hash setting, as specified in [RFC3174], which is used to look up the encryption certificate in the certificate **store** on the user's computer. The certificate that matches this hash SHOULD be used when sending encrypted mail to the **contact (3)**.

For a specified **PtypBinary** ([MS-OXCDATA] section 2.11.1) value in the **PidTagUserX509Certificate** property (section 2.2.4.66), if the **KeyExSHA1Hash** setting is present, the **KeyExchangeCertificate** setting MUST NOT be present.

The client and the server SHOULD use the **KeyExchangeCertificate** setting instead of the **KeyExSHA1Hash** setting.

### 2.2.4.66.7 SignSHA1Hash

The **TAG** value for this optional setting is 0x0009. The **LENGTH** value for this setting MUST be larger than 4. The **DATA** field contains the SHA1 Hash setting, as specified in [RFC3174], which is used to look up the signing certificate in the certificate store on the user's computer.<23>

For a specified **PtypBinary** ([MS-OXCDATA] section 2.11.1) value in the **PidTagUserX509Certificate** property (section 2.2.4.66), if the **SignSHA1Hash** setting is present, the **SignCertificate** setting MUST NOT be present.

The client and the server SHOULD use the **SignCertificate** setting instead of the **SignSHA1Hash** setting.

### 2.2.4.66.8 KeyExchangeCertificate

The **TAG** value for this setting is 0x0003. The **LENGTH** value for this setting MUST be larger than 4. The **DATA** field contains an ASN.1 DER encoded X.509 certificate, as specified in [RFC3280]. This

certificate SHOULD be used when sending encrypted S/MIME (Secure/Multipurpose Internet Mail Extensions) mail to the contact (3).

For a specified **PtypBinary** ([MS-OXCDATA] section 2.11.1) value in the **PidTagUserX509Certificate** property (section 2.2.4.66), if the **KeyExchangeCertificate** setting is present, the **KeyExSHA1Hash** setting MUST NOT be present. Conversely, if **KeyExSHA1Hash** is not present, the KeyExchangeCertificate setting MUST be present.

The client and the server SHOULD use the **KeyExchangeCertificate** setting instead of the **KeyExSHA1Hash** setting.

### 2.2.4.66.9 SignCertificate

The **TAG** value for this optional setting is 0x0008. The **LENGTH** value for this setting MUST be larger than 4. The **DATA** field contains an ASN.1 DER encoded X.509 certificate, as specified in [RFC3280].<24>

For a specified **PtypBinary** ([MS-OXCDATA] section 2.11.1) value in the **PidTagUserX509Certificate** property (section 2.2.4.66), if the **SignCertificate** setting is present, the **SignSHA1Hash** setting MUST NOT be present.

The client and the server SHOULD use the **SignCertificate** setting instead of the **SignSHA1Hash** setting.

### 2.2.4.66.10 ChainCertificate

The **TAG** value for this optional setting is 0x0004. The **LENGTH** value for this setting MUST be larger than 4. This setting can appear multiple times in the **PtypBinary** ([MS-OXCDATA] section 2.11.1) value to provide multiple certificates to use in the building of chains. The **DATA** field contains an ASN.1 DER encoded X.509 certificate, as specified in [RFC3280]. If present, these optional certificates SHOULD be used to build the chain of trust for the certificate specified by the **KeyExchangeCertificate** setting.

### 2.2.4.66.11 AsymetricCapabilities

The **TAG** value for this setting is 0x0002. The **LENGTH** value for this setting MUST be equal to or larger than 4. The **DATA** field contains the ASN.1 DER encoded **sMIMECapabilities** type, as specified in [RFC5751]. This setting MUST be part of the **PtypBinary** ([MS-OXCDATA] section 2.11.1) value.

### 2.2.4.66.12 SavedTime

The **TAG** value for this optional setting is 0x000D. The **LENGTH** value for this setting MUST be 0x000C. The **DATA** field MUST contain a **PtypFloatingTime** value that specifies the time (in UTC format) when the certificate was added to the contact (3).

### 2.2.4.67 PidTagAddressBookHomeMessageDatabase

Data type: **PtypString8** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookHomeMessageDatabase** property ([MS-OXPROPS] section 2.614) contains the DN (1) of a mail user, expressed in the *x500-dn* format specification described in section 2.2.1.1, of an Address Book object that represents the mail user's message database (store). The *x500-container-dn* portion of the DN (1) is an *x500-dn* string that references an Address Book object that represents the mail user's mailbox server. The RDN (2) in the *object-rdn*

of the *x500-dn* for the mailbox server's Address Book object is the host-name, as specified in [RFC1034], of the server that contains the mail user's message database.

### 2.2.4.68 PidTagAddressBookNetworkAddress

Data type: **PtypMultipleString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookNetworkAddress** property ([MS-OXPROPS] section 2.623) contains the Address Book object of a mailbox server that contains a list of names by which a server is known to the various transports in use by the network. Each **PtypString** is a remote procedure call (RPC) protocol sequence, as specified in [MS-RPCE], followed by a colon (":"), followed by the host-name of the server under that RPC protocol sequence.

### 2.2.4.69 PidTagHobbies

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagHobbies** property ([MS-OXPROPS] section 2.789) contains the names of a mail user's hobbies.

### 2.2.4.70 PidTagProfession

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagProfession** property ([MS-OXPROPS] section 2.938) contains the name of a mail user's line of business.

### 2.2.4.71 PidTagReferredByName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagReferredByName** property ([MS-OXPROPS] section 2.976) contains the name of a mail user's referral.

### 2.2.4.72 PidTagSpouseName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagSpouseName** property ([MS-OXPROPS] section 2.1085) contains the name of a mail user's spouse/partner.

### 2.2.4.73 PidTagGender

Data type: **PtypInteger16** ([MS-OXCDATA] section 2.11.1)

The **PidTagGender** property ([MS-OXPROPS] section 2.780) contains a value that represents a mail user's gender. If this value is present, it MUST contain one of the values listed in the following table.

| Value | Description |
|---|---|
| 0x0000 | No gender is specified. |
| 0x0001 | Specifies a gender of female. |
| 0x0002 | Specifies a gender of male. |

### 2.2.4.74 PidTagComputerNetworkName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagComputerNetworkName** property ([MS-OXPROPS] section 2.710) contains the name of a mail user's computer network.

### 2.2.4.75 PidTagCustomerId

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagCustomerId** property ([MS-OXPROPS] section 2.728) contains a mail user's customer identification number.

### 2.2.4.76 PidTagFtpSite

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagFtpSite** property ([MS-OXPROPS] section 2.778) contains a mail user's **File Transfer Protocol (FTP)** site address, as specified in [RFC959].

### 2.2.4.77 PidTagPersonalHomePage

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagPersonalHomePage** property ([MS-OXPROPS] section 2.927) contains the **Uniform Resource Locator (URL)** of a mail user's personal home page.

### 2.2.4.78 PidTagBusinessHomePage

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagBusinessHomePage** property ([MS-OXPROPS] section 2.695) contains the URL of a mail user's business home page.

### 2.2.4.79 PidTagBirthday

Data type: **PtypTime** ([MS-OXCDATA] section 2.11.1)

The **PidTagBirthday** property ([MS-OXPROPS] section 2.686) contains the date of a mail user's birthday at 12:00 a.m. Coordinated Universal Time (UTC).

### 2.2.4.80 PidTagWeddingAnniversary

Data type: **PtypTime** ([MS-OXCDATA] section 2.11.1)

The **PidTagWeddingAnniversary** property ([MS-OXPROPS] section 2.1126) contains the date of a mail user's wedding anniversary at 12:00 A.M. Coordinated Universal Time (UTC).

### 2.2.4.81 PidTagAddressBookOrganizationalUnitRootDistinguishedName

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookOrganizationalUnitRootDistinguishedName** property ([MS-OXPROPS] section 2.626) contains the DN (1) of the Organization object of the mail user's organization. When

this property is not present, clients obtain the mail user's organization as described in section 2.2.7.<25>

### 2.2.4.82   PidTagThumbnailPhoto

Data type: **PtypBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagThumbnailPhoto**<26> property ([MS-OXPROPS] section 2.1107), if present, MUST contain an image of a mail user's photo in Joint Photographic Experts Group (JPEG) format.

### 2.2.4.83   PidTagSpokenName

Data type: **PtypBinary** ([MS-OXCDATA] section 2.11.1)

The **PidTagSpokenName** property<27> ([MS-OXPROPS] section 2.1084), if present, MUST contain an Advanced Systems Format binary media object, as specified in [ASF] section 9.5, that contains a recording of the mail user's name pronunciation. The major media type MUST be set to 73647561-0000-0010-8000-00AA00389B71, the media subtype MUST be set to 00000001-0000-0010-8000-00AA00389B71, and the format type MUST be set to 05589f81-c356-11ce-bf01-00aa0055595a.

### 2.2.4.84   PidTagAddressBookAuthorizedSenders

Data type: **PtypObject** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookAuthorizedSenders** property<28> ([MS-OXPROPS] section 2.581) indicates whether delivery restrictions exist for the specified recipient (2). A value other than NULL indicates that delivery restrictions exist for this recipient (2). The address book does not contain the lists of senders who are allowed for this recipient (2); it only indicates whether such restrictions exist.

### 2.2.4.85   PidTagAddressBookUnauthorizedSenders

Data type: **PtypObject** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookUnauthorizedSenders** property<29> ([MS-OXPROPS] section 2.644) indicates whether delivery restrictions exist for the specified recipient (2). A value other than NULL indicates that delivery restrictions exist for this recipient (2). The address book does not contain the lists of senders who are prohibited for this recipient (2); it only indicates whether such restrictions exist.

### 2.2.4.86   PidTagAddressBookDistributionListMemberSubmitAccepted

Data type: **PtypObject** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookDistributionListMemberSubmitAccepted** property<30> ([MS-OXPROPS] section 2.588) indicates whether delivery restrictions exist for the specified recipient (2). A value other than NULL indicates that delivery restrictions exist for this recipient (2). The address book does not contain the lists of the group of senders who are allowed for this recipient (2); it only indicates whether such restrictions exist.

### 2.2.4.87   PidTagAddressBookDistributionListMemberSubmitRejected

Data type: **PtypObject** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookDistributionListMemberSubmitRejected** property<31> ([MS-OXPROPS] section 2.589) indicates whether delivery restrictions exist for this recipient (2). A value other than NULL indicates that delivery restrictions exist for this recipient (2). The address book does not contain the lists of the group of senders who are prohibited for this recipient (2); it only indicates whether such restrictions exist.

## 2.2.5 Properties That Reference Other Address Book Objects

***Note*** *Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

Some Address Book objects contain references to other Address Book objects, in the form of a table, through various properties.

For details about accessing these tables from an NSPI server, see [MS-NSPI] and [MS-OXNSPI] section 3.1.4.4.2.2.<32> Each of these properties is of type **PtypEmbeddedTable** ([MS-OXCDATA] section 2.11.1). For example, a mail user in an organization reports to a manager, who is also listed in that address book as another mail user. In this case, a property of type **PtypEmbeddedTable** or **PidTagAddressBookManager** (section 2.2.5.1) references the manager. In general, a property of type **PtypEmbeddedTable** references any number of other Address Book objects. Properties of type **PtypEmbeddedTable** are made available on an NSPI server by using a property value–based explicit table, through the **NspiGetMatches** method call, and are modified through the **NspiModLinkAtt** method call, as specified in [MS-NSPI], [MS-OXNSPI] section 3.1.4.1.10, and [MS-OXNSPI] section 3.1.4.1.15.<33>

These tables are represented in an offline address book (OAB) as a property either of type **PtypString** ([MS-OXCDATA] section 2.11.1) (for tables that consist of no more than one reference) or of type **PtypMultipleString** ([MS-OXCDATA] section 2.11.1) (for tables that can consist of multiple references), as specified in [MS-OXOAB]. Each string value represents the DN (1) of another Address Book object. For example, a mail user in an organization reports to a manager who is also listed in the address book as another mail user. In this case, the **PidTagAddressBookManager** property (section 2.2.5.1), with a **property type** of **PtypString**, contains the DN (1) of the manager.

### 2.2.5.1 PidTagAddressBookManager

Data type: **PtypObject** ([MS-OXCDATA] section 2.11.1.5)

The **PidTagAddressBookManager** property ([MS-OXPROPS] section 2.618) contains one row that references a mail user's manager.

### 2.2.5.2 PidTagAddressBookReports

Data type: **PtypEmbeddedTable** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookReports** property ([MS-OXPROPS] section 2.637) lists all of the mail user's direct reports.

### 2.2.5.3 PidTagAddressBookIsMemberOfDistributionList

Data type: **PtypString8** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookIsMemberOfDistributionList** property ([MS-OXPROPS] section 2.616) is an Address Book object that lists all the distribution lists for which a specified object is a member.

### 2.2.5.4 PidTagAddressBookOwnerBackLink

Data type: **PtypEmbeddedTable** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookOwnerBackLink** property ([MS-OXPROPS] section 2.628) is a mail user list that contains the distribution lists that the mail user owns.

### 2.2.5.5 PidTagAddressBookPublicDelegates

Data type: **PtypEmbeddedTable** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookPublicDelegates** property ([MS-OXPROPS] section 2.636) contains a list of mail users who are allowed to send mail on behalf of the mailbox owner.

### 2.2.5.6 PidTagAddressBookHierarchicalShowInDepartments

Data type: **PtypEmbeddedTable** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookHierarchicalShowInDepartments** property ([MS-OXPROPS] section 2.613) is a mail user list that contains all of the Department objects of which the mail user is a member.

## 2.2.6 Properties That Apply to Distribution Lists

### 2.2.6.1 PidTagAddressBookMember

Data type: **PtypEmbeddedTable** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookMember** property ([MS-OXPROPS] section 2.620) contains the members of a distribution list. If the distribution list is also a departmental group (as specified by the **PidTagAddressBookHierarchicalIsHierarchicalGroup** property (section 2.2.6.5)), the **PidTagAddressBookMember** property (section 2.2.6.1) contains the members of the department and the child departmental groups in the hierarchy of departments.

### 2.2.6.2 PidTagAddressBookOwner

Data type: **PtypEmbeddedTable** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookOwner** property ([MS-OXPROPS] section 2.627) contains one row that references a distribution list's owner.

### 2.2.6.3 PidTagContainerContents

Data type: **PtypEmbeddedTable** ([MS-OXCDATA] section 2.11.1)

The **PidTagContainerContents** property ([MS-OXPROPS] section 2.713) contains a distribution list that is always empty. An NSPI server MUST define this value for distribution lists. This value is not present for any other objects.

### 2.2.6.4 PidTagAddressBookFolderPathname

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookFolderPathname** property ([MS-OXPROPS] section 2.607) has been deprecated and MUST be ignored by clients.

### 2.2.6.5 PidTagAddressBookHierarchicalIsHierarchicalGroup

Data type: **PtypBoolean** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookHierarchicalIsHierarchicalGroup** property ([MS-OXPROPS] section 2.610) contains **TRUE** (0x01) if the distribution list represents a departmental group. Members of this departmental group that are also distribution lists with the **PidTagAddressBookHierarchicalIsHierarchicalGroup** property (section 2.2.6.5) set to **TRUE** represent the child department of this departmental group in a hierarchy of departments. Other members represent members of this departmental group. If the distribution list is not a departmental group, the **PidTagAddressBookHierarchicalIsHierarchicalGroup** property can be either **FALSE** (0x00) or absent.

## 2.2.7 Properties That Apply to Organization Objects

An Organization object is an Address Book object that represents an organization and contains properties that are specific to that organization. A mail user's Organization object is obtained by using the **PidTagAddressBookOrganizationalUnitRootDistinguishedName** property (section 2.2.4.81) of the mail user.<34> When this property is not present on a mail user, messaging clients access the Organization object of a mail user by extracting the *org-rdn* string from the mail user's DN (1), which is in the format of an X500 address and the *x500-dn* format specification, as specified in section 2.2.1.1. Messaging clients use the resulting *org-rdn* string as a DN (1) of the mail user's organization, which is in the *organization-dn* format specification. After the Organization object is obtained, messaging clients are able to perform operations the same way as they would on any other Address Book object.

### 2.2.7.1 PidTagAddressBookRoomContainers

Data type: **PtypMultipleString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookRoomContainers** property ([MS-OXPROPS] section 2.639) contains a list of DNs (1) representing the address book containers that hold Resource objects, such as conference rooms and equipment. Messaging clients use this list to determine which containers have mainly Resource objects so that they can perform special actions on these containers, such as displaying a different column set when browsing address lists that are represented by these containers, or using features that require selecting from a set of rooms or equipment.

The DNs (1) in the **PidTagAddressBookRoomContainers** property each use the *x500-dn* format specification, as specified in section 2.2.1.1, with the additional requirement that there is no *container-rdn* in its *x500-dn* and that the RDN (2) of the *object-rdn* follows the *container-guid* format specification. When the RDN (2) is extracted from the DN (1), it represents the GUID of the address book container that this is referencing. Messaging clients compare this GUID to the GUID of another address book container by extracting that GUID from the *container-guid* of the other container's DN (1), which follows the *addresslist-dn* format specification. If the **GUIDs** are the same, the other address book container is a room container.

### 2.2.7.2 PidTagAddressBookHierarchicalRootDepartment

Data type: **PtypEmbeddedTable** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookHierarchicalRootDepartment** property ([MS-OXPROPS] section 2.612) contains a reference to either the root Department object or the root departmental group in the department hierarchy for an organization.<35> The table has either zero rows or one row, which references a Department object or departmental group. If either the Organization object is missing,

this property is missing, or the property value is empty, the NSPI server does not have a department hierarchy for that organization.

## 2.2.8  Properties That Apply to Department Objects

*Note*  *Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

A Department object is an Address Book object that represents a department within an organization and contains properties that are specific to a Department object. Department objects are distinct from departmental groups, which are a specific type of distribution list with all its properties. A server MAY<36> represent a department hierarchy by using either Department objects or departmental groups, and it MUST NOT mix the representations.

Messaging clients obtain the root of the department hierarchy by using the **PidTagAddressBookHierarchicalRootDepartment** property (section 2.2.7.2) of the Organization object (which MAY be either a Department object or a departmental group), or by obtaining the root department that is not specific to any organization, which MUST be a Department object, and which has a DN (1) that is specified by using the *organization-dn* format specification, as specified in 2.2.1.1, with a value of "/o=FF46312B-D8AE-406C-B8E6-BC1A22A4C69E".

### 2.2.8.1  PidTagAddressBookHierarchicalChildDepartments

Data type: **PtypEmbeddedTable** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookHierarchicalChildDepartments** property ([MS-OXPROPS] section 2.608) contains a Department object that references the child departments in a hierarchy of departments.

### 2.2.8.2  PidTagAddressBookHierarchicalParentDepartment

Data type: **PtypEmbeddedTable** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookHierarchicalParentDepartment** property ([MS-OXPROPS] section 2.611) contains a Department object that references all of the departments to which this specified department is a child department.

### 2.2.8.3  PidTagAddressBookHierarchicalDepartmentMembers

Data type: **PtypEmbeddedTable** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookHierarchicalDepartmentMembers** property ([MS-OXPROPS] section 2.609) contains a Department object that lists all of the mail users that belong to a specified department.

## 2.2.9  Properties That Apply to Resource Objects

### 2.2.9.1  PidTagAddressBookRoomCapacity

Data type: **PtypInteger32** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookRoomCapacity** property ([MS-OXPROPS] section 2.638) contains the maximum occupancy of a specified room.

### 2.2.9.2 PidTagAddressBookRoomDescription

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookRoomDescription** property ([MS-OXPROPS] section 2.640) contains a description of a Resource object.

## 2.2.10 Properties That Have Special Purposes

### 2.2.10.1 PidTagAnr

*Note   Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

Data type: **PtypString** ([MS-OXCDATA] section 2.11.1)

The **PidTagAnr** property ([MS-OXPROPS] section 2.648) is a special property that is not actually a property on the Address Book objects themselves. Rather, this property is used by messaging clients as a property value to a *Filter* parameter to the **NspiGetMatches** method call, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.10.<37> Messaging clients pass this property as a target string to the **NspiGetMatches** method call to identify objects in an address list that are possible matches for the target string. This operation is known as **ambiguous name resolution (ANR)**, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.7.<38> NSPI servers respond by returning the Minimal Entry IDs of all Address Book objects that are possible matches against the target string. This protocol does not prescribe the choice of ANR results of an NSPI server.<39>

An offline address book (OAB) does not contain this property for any Address Book objects. It is up to messaging clients to determine how to perform name matching among the objects in an OAB.

### 2.2.10.2 PidTagAddressBookManageDistributionList

Data type: **PtypObject** ([MS-OXCDATA] section 2.11.1)

The **PidTagAddressBookManageDistributionList** property ([MS-OXPROPS] section 2.617) is a **property tag** for use in display templates for distribution lists. When the value for the **PidTagAddressBookManageDistributionList** property is the **dwType** property tag in a button control to a **CNTRL** structure of a display template, it tells messaging clients to include a button in the user interface to enable them to edit the members of a distribution list. For details about the button control for a template, see [MS-OXOABKT]. This is not a property of objects in an address book.

## 2.2.11 Named Properties

This document does not specify any **named properties**. OABs and NSPI servers can expose any named properties in their implementations.

# 3   Protocol Details

## 3.1   Client Details

### 3.1.1   Abstract Data Model

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

All abstract data model (ADM) types maintained by the client are prefixed with "Client".

The following ADM types are defined in this section:

**Client.AddressBook:** A collection of **Client.AddressBookObject** ADM types, each of which are contained in any number of **AddressList** ADM types. Among the many types of objects, a **Client.AddressBook** ADM type includes the following object types:

- Mailbox user, each of which is a mailbox owner.

- Mail users, each of which describes a person or entity that can receive e-mail messages.

- Distribution lists, each of which is a collection of other mail users, distribution lists, or other **Client.AddressBookObject** ADM types that can receive e-mail messages.

- Resource objects that can be reserved, such as rooms or equipment.

- Organization objects, each of which describes an organization.

- Department objects, which describe the departmental structure of an organization.

- Templates, each of which describes a physical view that can be used to show details on other Address Book objects to a messaging user, as specified in [MS-OXOABKT].

- **Client.AddressBookContainers** ADM types.

**Client.AddressBookContainer:** A **Client.AddressBookObject** ADM type that describes a **Client.AddressList** ADM type.

**Client.AddressBookHierarchyTable:** A collection of **Client.AddressBookContainer** ADM types arranged in a hierarchy. Messaging users are able to scroll through this table of objects, set positioning based on a name typed by a messaging user, perform searches, and perform ambiguous name resolution (ANR) against the names in that **Client.AddressList** ADM type. Messaging users are able to display information about a **Client.AddressBookObject** ADM type from the table, an object obtained through ANR, or a recipient (2) on a message. The information displayed can be formatted according to a display template that is supplied in the **Client.AddressBookObject** ADM type, and the type of information depends on the type of **Client.AddressBookObject** ADM type being displayed.

**Client.AddressBookObject:** An entity in a **Client.AddressBook** ADM type that contains a set of **attributes (1)**, each attribute (1) with a set of associated values

**Client.AddressList:** A collection of distinct **Client.AddressBookObject** ADM types.

**Client.OfflineAddressBook:** A collection of **Client.AddressList** ADM types that are stored in a format that a client can save and use locally.

### 3.1.2  Timers

None.

### 3.1.3  Initialization

Initialization is accomplished in one of three ways: through a name service provider interface (NSPI) connection to a server, through the contents of an offline address book (OAB), or both.

#### 3.1.3.1  Initialization Through NSPI connection

*Note  Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

Initialization is accomplished by means of the **NspiBind** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.1,<40> and MUST occur before any Address Book objects can be accessed through NSPI calls.

#### 3.1.3.2  Initialization Through an Offline Address Book

*Note  Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

Messaging clients MAY<41> obtain address lists in an OAB by using the OAB Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB]. Each address list, in turn, contains information about objects in that address list, using the OAB Format and Schema Protocol, as specified in [MS-OXOAB]. After one or more address lists from an OAB are obtained, messaging clients navigate any of the address lists to perform the lookups that are required to perform these operations.

### 3.1.4  Higher-Layer Triggered Events

It is possible for messaging clients to obtain a hierarchy of address book containers, browse the Address Book objects in an address list that is represented by an address book container, obtain information about an Address Book object, and perform ambiguous name resolution (ANR) to find Address Book objects that match a specified target string.

#### 3.1.4.1  Obtaining a Hierarchy of Address Book Containers

*Note  Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

When using an NSPI server, messaging clients obtain the address book hierarchy table by using the **NspiGetSpecialTable** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.3.<42> The hierarchy table is a set of rows, each of which describes one address book container.

The hierarchy table in an OAB MAY<43> be obtained by using the OAB Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB].

### 3.1.4.2   Browsing an Address Book

***Note***   *Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

After the hierarchy of containers is obtained and a single container is chosen by a messaging user, it is possible for the messaging user to browse the address list that is represented by the container.

When browsing an address list for an OAB, it is up to the messaging client to perform the lookups that are necessary to present a browsable view of the address list, by using the OAB Format and Schema Protocol, as specified in [MS-OXOAB].

When browsing an address list from an NSPI server, messaging clients use the **NspiQueryRows** method to obtain a set of rows to display to the messaging user. Messaging clients choose the properties they want to render, minimally they SHOULD request the **PidTagEntryId** property (section 2.2.3.2), the **PidTagDisplayName** property (section 2.2.3.1), the **PidTagSmtpAddress** property (section 2.2.3.21), and the **PidTagTitle** property (section 2.2.4.7), among other properties that the messaging client deems useful for displaying to the user.

It is possible to scroll to certain approximate positions in the address book view. When using NSPI, this is accomplished by modifying the **STAT** structure, which describes a table position, as specified in [MS-NSPI] and [MS-OXNSPI] section 2.3.7,<44> and then calling the **NspiUpdateStat** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.4.<45> Messaging clients follow up by calling the **NspiQueryRows** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.8, to display rows starting at the new position.

It is possible for the user to type a particular name and have the address list view scroll to the first display name typed, like a Rolodex. When using NSPI, this is accomplished through the **NspiSeekEntries** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.9. The result updates positioning information in the **STAT** structure, as specified in [MS-NSPI] and [MS-OXNSPI] section 2.3.7, and returns a screen full of Address Book objects.

### 3.1.4.3   Obtaining Properties on an Address Book Object

***Note***   *Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

To obtain properties for an Address Book object from an OAB, the messaging client needs to have either a DN (1) for the object or the object's Simple Mail Transfer Protocol (SMTP) address. This means that the messaging client will need to keep track of the DN (1) after any lookup in the OAB, and it will need to make sure that the DN (1) or SMTP address is maintained for the addressee of a message when messages are sent or received through a messaging server. Incoming messages can contain either the SMTP address or the DN (1) of message addressees. The messaging server provides the SMTP address or the DN (1) of the addressee. The OAB Format and Schema Protocol Specification, as specified in [MS-OXOAB], describes how this information is organized and structured. After the information for the Address Book object is located, it is possible for the client to retrieve the value of any property in that record. For example, to obtain properties such as the business telephone number and SMTP address, the client looks up the **PidTagBusinessTelephoneNumber** property (section 2.2.4.38) and the **PidTagSmtpAddress** property (section 2.2.3.21) in the record.

To obtain properties for an Address Book object on an NSPI server, the messaging client needs to either have a DN (1), a Minimal Entry ID or the object's SMTP address. Specific NSPI calls that return Address Book object information — including but not limited to:

- The **NspiQueryRows** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.8<46>

- The **NspiSeekEntries** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.9

- The **NspiGetMatches** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.10

— include the DN (1) or Minimal Entry ID in their calls for the **PidTagEntryId** property, or as a return value of the calls. Clients call the **NspiDNToMId** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.13, to obtain the Minimal Entry ID from a DN (1). To obtain a DN (1) from an SMTP address, messaging clients construct a string that starts with "=SMTP: " followed by the SMTP address as the *paStr* or *paWStr* parameter to either the **NspiResolveNames** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.16, or the **NspiResolveNamesW** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.17, and extract the DN (1) from the **PidTagEntryId** property (section 2.2.3.2) that is returned, as specified in the **PermanentEntryID** structure, as specified in [MS-NSPI] and [MS-OXNSPI] section 2.3.8.3.<47>

After the Minimal Entry ID is known, clients use it as the **CurrentRec** member of the **STAT** structure passed to the **NspiGetProps** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.7. It is possible for clients to specify any number of properties in the *pPropTags* parameter to the **NspiGetProps** method that the client wants to request. For example, to request properties such as the business phone number and SMTP address, the client includes the **property IDs** for the **PidTagBusinessTelephoneNumber** (section 2.2.4.38) and **PidTagSmtpAddress** (section 2.2.3.21) properties in the *pPropTags* parameter.

### 3.1.4.4   Performing Ambiguous Name Resolution

*Note*   *Some of the information in this section is subject to change because it applies to a preliminary implementation of the protocol or structure. For information about specific differences between versions, see the behavior notes that are provided in the Product Behavior appendix.*

Messaging clients perform ambiguous name resolution (ANR) when a user is addressing a message and types only part of a name. The role of the address book in this case is to identify the best possible matches for the name entered, and if there is more than one good match, to present the list of possible Address Book objects to the user.

When performing ANR by using an offline address book (OAB), it is up to the messaging client to decide the best method for name matching among Address Book objects. The client is free to use any available means to determine good matches on any choice of properties by using the OAB Format and Schema Protocol, as specified in [MS-OXOAB].

When performing ANR by using an NSPI server, the client calls the **NspiResolveNames** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.16,<48> or the **NspiResolveNamesW** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.17, and passes the target string in the *paStr* or *paWStr* parameter. If there is exactly one match, the row that corresponds to the target string will contain the properties that were requested for that match. If there is more than one match, it is possible to obtain the set of possible matches by calling the **NspiGetMatches** method, as specified in [MS-NSPI] and [MS-OXNSPI] section 3.1.4.1.10, by using a *Filter* parameter with the **PidTagAnr** property and using the value for the **PidTagAnr** property as the target string for ANR. The set of matches, along with the requested properties for the matches, will be returned. Messaging clients display the returned results in a dialog box, so the user is able to select the best match from the list of results.

### 3.1.5   Message Processing Events and Sequencing Rules

None.

### 3.1.6 Timer Events

None.

### 3.1.7 Other Local Events

None.

## 3.2 Server Details

### 3.2.1 Abstract Data Model

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

All ADM types maintained by the server are prefixed with "Server".

The following ADM types are defined in this section:

**Server.AddressBook:** A collection of **Server.AddressBookObject** ADM types, each of which are contained in any number of **Server.AddressList** ADM types. Among the many types of objects, a **Server.AddressBook** ADM type includes the following object types:

- Mailbox user, each of is a mailbox owner.

- Mail users, each of which describes a person or entity that can receive e-mail messages.

- Distribution lists, each of which is a collection of other mail users, distribution lists, or other **Server.AddressBookObject** ADM types that can receive e-mail messages.

- Resource objects that can be reserved, such as rooms or equipment.

- Organization objects, each of which describes an organization.

- Department objects, which describe the departmental structure of an organization.

- Templates, each of which describe a physical view that can be used to show details on other Server.AddressBookObjects to a messaging user, as specified in [MS-OXOABKT].

- **Server.AddressBookContainer** ADM types.

**Server.AddressBookContainer:** A **Server.AddressBookObject** ADM type that describes a **Server.AddressList** ADM type.

**Server.AddressBookHierarchyTable:** A collection of **Server.AddressBookContainer** ADM types arranged in a hierarchy. Messaging users are able to scroll through this table of objects, set positioning based on a name typed by a messaging user, perform searches, and perform ambiguous name resolution (ANR) against the names in that **Server.AddressList** ADM type. Messaging users are able to display information about a **Server.AddressBookObject** ADM type from the table, an object obtained through ANR, or a recipient (2) on a message. The information displayed can be formatted according to a display template that is supplied in the **Server.AddressBookObject** ADM type, and the type of information depends on the type of **Server.AddressBookObject** ADM type being displayed.

**Server.AddressBookObject:** An entity in a **Server.AddressBook** ADM type that contains a set of attributes (1), each attribute (1) with a set of associated values

**Server.AddressList:** A collection of distinct **Server.AddressBookObject** ADM types.

**Server.OfflineAddressBook:** A collection of **Server.AddressList** ADM types that are stored in a format that a client can save and use locally.

### 3.2.2   Timers

None.

### 3.2.3   Initialization

None.

### 3.2.4   Higher-Layer Triggered Events

None.

### 3.2.5   Message Processing Events and Sequencing Rules

An OAB does not contain the following properties, but the server MUST generate values for these properties before sending the OAB to the client:

- **PidTagDepth** (section 2.2.2.2)

- **PidTagContainerFlags** (section 2.2.2.1)

### 3.2.6   Timer Events

None.

### 3.2.7   Other Local Events

None.

# 4   Protocol Examples

## 4.1   Call Sequence To Retrieve Address Book Properties

This section describes the call sequence for obtaining two string properties — **PidTagDisplayName** (section 2.2.3.1) and **PidTagGivenName** (section 2.2.4.2) — for a mail user whose DN (1) is "/o=First Organization/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/CN=Recipients/CN=user1." The following figure shows the call sequence between the client and the server.
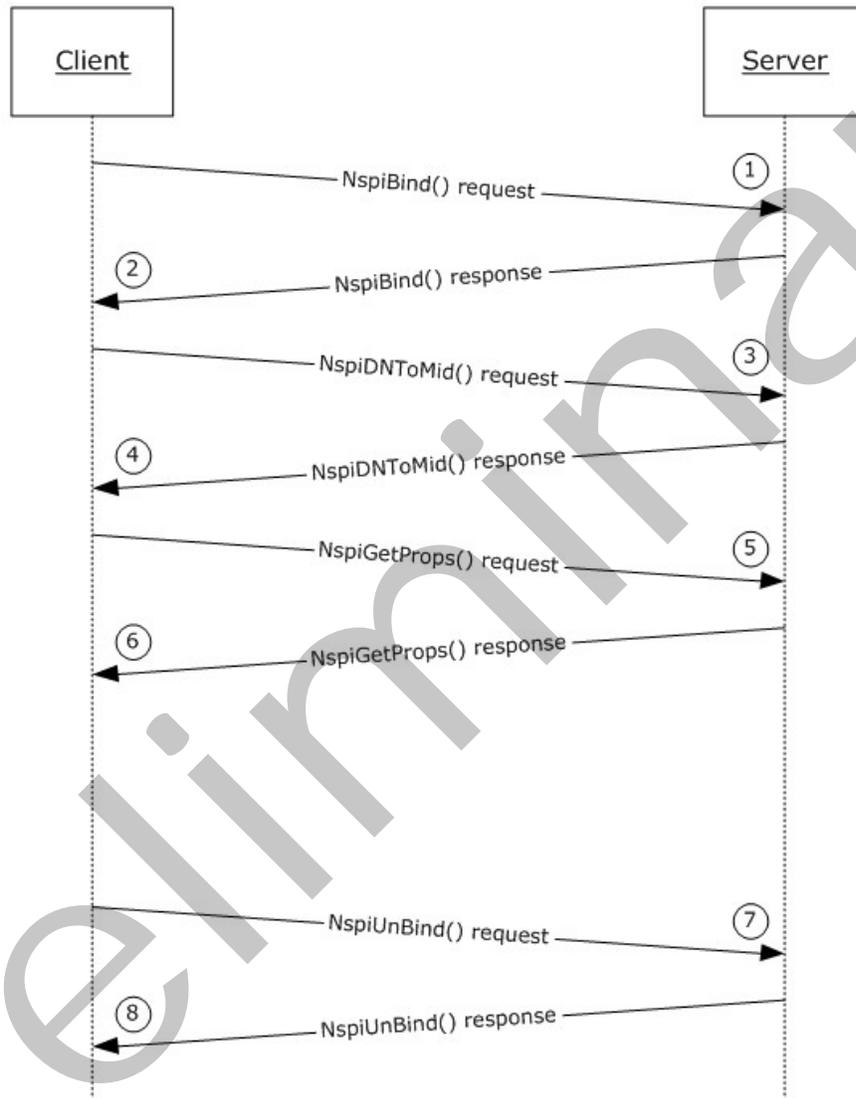
**Figure 1: Call sequence between client and server**

**Note** This figure shows relevant information only, not all of the possible parameters. For more details about the parameters, see [MS-NSPI] and [MS-OXNSPI].

1. The client initiates a session to the NSPI server by calling the **NspiBind** method. The messaging client sends in the following values to the server:

```
dwFlags
             0x00000000           DWORD
pStat
             hIndex            0x00000000              unsigned long
             ContainerID       0x00000000              unsigned long
             CurrentRec        0x00000000              unsigned long
             Delta             0x00000000              long
             NumPos            0x00000000              unsigned long
             TotalRecs         0x00000000              unsigned long
             CodePage          0x000004e4              unsigned long
             TemplateLocale    0x00000409              unsigned long
             SortLocale        0x00000409              unsigned long
pServerGuid
      < a pointer to an array of 16 unsigned chars to be returned by the server>
```

2. The server responds to the **NspiBind** method call with the return code "Success" and a valid server GUID. Typical parameters are as follows:

```
pServerGuid
             [0x0]    0xab 0xbc 0x8b 0x86 0x79 0x33 0xc4        0x48 0xa1 0xef
             [0xa]    0x1b 0x53 0xe6 0x3b 0xdc 0x46
contextHandle
             < a token that will be used by the NspiUnbind call>
```

3. The client requests the Minimal Entry ID for the DN (1) "/o=First Organization/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/CN=Recipients/CN=user1" by calling the **NspiDNToMId** method with the following parameters:

```
_pNames
             Count      0x00000001           DWORD
             Strings                         char **
                     [0x0]                   char *
             "/o=First Organization/ou=Exchange Administrative Group
             (FYDIBOHF23SPDLT)/cn=Recipients/cn=user1"
ppMIds
        <a pointer to a valid memory location of type PropertyTagArray_r ** for server
to return Mid>
```

4. The server responds with return code of **Success** with a value of 0x00001927 for a Minimal Entry ID as follows:

```
             Count      0x00000001              DWORD
             Strings                            char **
                     [0x0]                      char *
                         "/o=First Organization/ou=Exchange Administrative
                         Group (FYDIBOHF23SPDLT)/cn=Recipients/cn=user1"
ppMIds
             cValues    0x00000001              DWORD
             aulPropTag                         DWORD[]
                     [0x0]
                     0x00001927                 DWORD
```

5. The client requests two string properties — **PidTagDisplayName** (section 2.2.3.1) and **PidTagGivenName** (section 2.2.4.2) — by calling the **NspiGetProps** method with the following parameters:

**Note** The Minimal Entry ID value of 0x00001927 that was obtained in step 4 is used as the **CurrentRec** field of the *pStat* parameter.

```
hRpc          < a valid RPC handle>                    void *
dwFlags       0x00000000            DWORD
pStat
              hIndex               0x00000000          unsigned long
              ContainerID          0x00000000          unsigned long
              CurrentRec           0x00001927          unsigned long
              Delta                0x00000000          long
              NumPos               0x00000000          unsigned long
              TotalRecs            0x00000000          unsigned long
              CodePage             0x000004b0          unsigned long
              TemplateLocale       0x00000409          unsigned long
              SortLocale           0x00000409          unsigned long
pPropTags            _SPropTagArray_r *
        {
                cValues           0x00000002          DWORD
                aulPropTag=<a pointer to an array of proptags>
        }
        aulPropTag                                    unsigned long []
                [0x0]             PidTagDisplayName   unsigned long
                [0x1]             PidTagGivenName     unsigned long
ppRow               _SRow_r * *
        < memory location for server return values>
```

6. The server responds to the **NspiGetProps** method call with the return code of **Success**. In this example, the server has returned the string value "user1" for both of the requested properties, and the return values are represented as follows:

```
dwFlags       0x00000000            DWORD
pStat
{
              hIndex               0x00000000          unsigned long
              ContainerID          0x00000000          unsigned long
              CurrentRec           0x00001927          unsigned long
              Delta                0x00000000          long
              NumPos               0x00000000          unsigned long
              TotalRecs            0x00000000          unsigned long
              CodePage             0x000004b0          unsigned long
              TemplateLocale       0x00000409          unsigned long
              SortLocale           0x00000409          unsigned long
}
pPropTags            _SPropTagArray_r *
{
                cValues           0x00000002          DWORD
                aulPropTag=<a pointer to an array of proptags>
}
        aulPropTag                                    unsigned long []
                [0x0]             PidTagDisplayName   unsigned long
                [0x1]             PidTagGivenName     unsigned long

ppRows      _SRowSet_r * *
```

```
                    {
                            cRows                 0x00000001           DWORD
                            aRow=<a pointer to an array of rows>
                    }
```

In this example, the server has returned a total of 0x1 row denoted as [0x0] that is represented as follows:

```
    aRow        [0x0]      _SRow_r  *
                    {
                            cValues               0x00000002           DWORD
                            lpProps=<a pointer to an array of columns>
                    }
```

In this example, the server has returned a column set of two properties, and each column will be represented as follows:

```
    [0x0]      _SPropValue_r
            {
            [0x0]
                            ulPropTag         PidTagDisplayName       unsigned long
                            Value
                                    lpszA =   "user1"char *
            [0x1]
                            ulPropTag         PidTagGivenName         unsigned long
                            Value
                                    lpszA =   "user1" char *
            }
```

**Note** The client can invoke additional NSPI calls to access other information from the server before calling the **NSPIUnbind** method.

7. The client terminates the connection by calling the **NspiUnbind** method with a token that the server returned in response to an **NspiBind** method call.

```
    contextHandle          NSPI_HANDLE   *
                  <a token which was sent by the server in the NspiBind call >
    dwFlags
                      0x00000000                              unsigned long
```

8. The server responds with the return code 0x00000001 and destroys the token that the client passed in.

## 4.2   PidTagUserX509Certificate Property in ASN.1 DER Encoded Format

The following example shows the parsed text output of a sample **PidTagUserX509Certificate** property (section 2.2.4.66) value in ASN.1 DER encoded **SignedData** type BLOB format.

The first column contains the offset to the node in bytes, the second column contains the length of the node in bytes, the third column contains the byte count of the hex value of the second column (for example, 2228 is DB 6D BC in hex, which is 3 bytes), and the fourth column contains the text output of the node contents.

```
   0|  2228|     3| SEQUENCE :
   4|     9|     1|    OBJECT IDENTIFIER : signedData [1.2.840.113549.1.7.2]
  15|  2213|     3|    CONTEXT SPECIFIC (0) :
  19|  2209|     3|       SEQUENCE :
  23|     1|     1|          INTEGER : 1
  26|    11|     1|          SET :
  28|     9|     1|             SEQUENCE :
  30|     5|     1|                OBJECT IDENTIFIER : sha1 [1.3.14.3.2.26]
  37|     0|     1|                NULL : ''
  39|    25|     1|          SEQUENCE :
  41|     9|     1|             OBJECT IDENTIFIER : data [1.2.840.113549.1.7.1]
  52|    12|     1|             CONTEXT SPECIFIC (0) :
  54|    10|     1|                OCTET STRING :
    |      |      |                   'Empty Body'
  66|  1509|     3|          CONTEXT SPECIFIC (0) :
  70|   638|     3|             SEQUENCE :
  74|   487|     3|                SEQUENCE :
  78|     3|     1|                   CONTEXT SPECIFIC (0) :
  80|     1|     1|                      INTEGER : 2
  83|    16|     1|                   INTEGER :
    |      |      |                      0274DE397E583BF0DB53CE0EECB024A0
 101|    13|     1|                   SEQUENCE :
 103|     9|     1|                      OBJECT IDENTIFIER : sha1withRSAEncryption
[1.2.840.113549.1.1.5]
 114|     0|     1|                      NULL : ''
 116|    86|     1|                   SEQUENCE :
 118|    60|     1|                      SET :
 120|    58|     1|                         SEQUENCE :
 122|     9|     1|                            OBJECT IDENTIFIER : emailAddress
[1.2.840.113549.1.9.1]
 133|    45|     1|                            IA5 STRING :
    |      |      |                               'Administrator@ssvsim-dom.extes'
    |      |      |                               't.microsoft.com'
 180|    22|     1|                      SET :
 182|    20|     1|                         SEQUENCE :
 184|     3|     1|                            OBJECT IDENTIFIER : commonName [2.5.4.3]
 189|    13|     1|                            UTF8 STRING :
    |      |      |                               'Administrator'
 204|    30|     1|                   SEQUENCE :
 206|    13|     1|                      UTC TIME :
    |      |      |                         '111108025113Z'
 221|    13|     1|                      UTC TIME :
    |      |      |                         '111208025113Z'
 236|    86|     1|                   SEQUENCE :
 238|    60|     1|                      SET :
 240|    58|     1|                         SEQUENCE :
 242|     9|     1|                            OBJECT IDENTIFIER : emailAddress
[1.2.840.113549.1.9.1]
 253|    45|     1|                            IA5 STRING :
    |      |      |                               'Administrator@ssvsim-dom.extes'
    |      |      |                               't.microsoft.com'
 300|    22|     1|                      SET :
 302|    20|     1|                         SEQUENCE :
 304|     3|     1|                            OBJECT IDENTIFIER : commonName [2.5.4.3]
 309|    13|     1|                            UTF8 STRING :
    |      |      |                               'Administrator'
 324|   159|     2|                   SEQUENCE :
 327|    13|     1|                      SEQUENCE :
```

```
 329|    9|    1|                                OBJECT IDENTIFIER : rsaEncryption
[1.2.840.113549.1.1.1]
 340|    0|    1|                                NULL : ''
 342|  141|    2|                            BIT STRING UnusedBits:0 :
    |     |     |
 346|  137|    2|                                SEQUENCE :
 349|  129|    2|                                    INTEGER :
    |     |     |                                        00921EA7244DD0D37C9107C2B46D7C
    |     |     |                                        2B8BE61B22E971488E559B47E122D8
    |     |     |                                        D255F248062BCB35FE3FEC0AE07EC3
    |     |     |                                        2FB59196C987B953EC28D68741BA26
    |     |     |                                        F794902FDC1AB9F7C8F33A9ED9398D
    |     |     |                                        4C363EC56D05E21EDC01DF3F01EB18
    |     |     |                                        C4B80087039B3D3C77E3792028EEF3
    |     |     |                                        635DBC70C811F327D3493DAA431E23
    |     |     |                                        69879056081F722827
 481|    3|    1|                                    INTEGER : 65537
 486|   77|    1|                            CONTEXT SPECIFIC (3) :
 488|   75|    1|                                SEQUENCE :
 490|   29|    1|                                    SEQUENCE :
 492|    3|    1|                                        OBJECT IDENTIFIER : subjectKeyIdentifier
[2.5.29.14]
 497|   22|    1|                                        OCTET STRING :
 499|   20|    1|                                            OCTET STRING :
    |     |     |                                                C38DC1AB5265B974F7A5DFBDE8E
    |     |     |                                                B75F60E2AEF8F
 521|   11|    1|                                    SEQUENCE :
 523|    3|    1|                                        OBJECT IDENTIFIER : keyUsage [2.5.29.15]
 528|    4|    1|                                        OCTET STRING :
 530|    2|    1|                                            BIT STRING UnusedBits:7 :
    |     |     |                                                80
 534|   29|    1|                                    SEQUENCE :
 536|    3|    1|                                        OBJECT IDENTIFIER : extKeyUsage [2.5.29.37]
 541|   22|    1|                                        OCTET STRING :
 543|   20|    1|                                            SEQUENCE :
 545|    8|    1|                                                OBJECT IDENTIFIER : clientAuth
[1.3.6.1.5.5.7.3.2]
 555|    8|    1|                                                OBJECT IDENTIFIER : emailProtection
[1.3.6.1.5.5.7.3.4]
 565|   13|    1|                    SEQUENCE :
 567|    9|    1|                        OBJECT IDENTIFIER : sha1withRSAEncryption
[1.2.840.113549.1.1.5]
 578|    0|    1|                        NULL : ''
 580|  129|    2|                    BIT STRING UnusedBits:0 :
    |     |     |                        2DEC723B68A73E04779D17854E8F9068E6E0ACD36F
    |     |     |                        ABDCB5DA3A8FCB12772547AED11A03B13F6358DED2
    |     |     |                        2CE1B568D70AED842A9B099C4B77B21ABC0E48F751
    |     |     |                        CC4251A8804472DD87A4EE04709923AD711A1C6267
    |     |     |                        5460EA27A380773542A1C248BCD30EA5BDEEA923D8
    |     |     |                        385523176DF5987D9C668E847B3A2D5C57CD22F693
    |     |     |                        2CF4
 712|  863|    3|            SEQUENCE :
 716|  712|    3|                SEQUENCE :
 720|    3|    1|                    CONTEXT SPECIFIC (0) :
 722|    1|    1|                        INTEGER : 2
 725|   10|    1|                    INTEGER :
    |     |     |                        7EC4FFE5000000001F5F
 737|   13|    1|                    SEQUENCE :
```

*58 / 73*

```
  739|      9|      1|                              OBJECT IDENTIFIER : sha1withRSAEncryption
[1.2.840.113549.1.1.5]
  750|      0|      1|                              NULL : ''
  752|     21|      1|                          SEQUENCE :
  754|     19|      1|                            SET :
  756|     17|      1|                              SEQUENCE :
  758|      3|      1|                                OBJECT IDENTIFIER : commonName [2.5.4.3]
  763|     10|      1|                                PRINTABLE STRING :
     |       |       |                                    'RSACERTSRV'
  775|     30|      1|                          SEQUENCE :
  777|     13|      1|                            UTC TIME :
     |       |       |                                '111108010318Z'
  792|     13|      1|                            UTC TIME :
     |       |       |                                '121108011318Z'
  807|     86|      1|                          SEQUENCE :
  809|     22|      1|                            SET :
  811|     20|      1|                              SEQUENCE :
  813|      3|      1|                                OBJECT IDENTIFIER : commonName [2.5.4.3]
  818|     13|      1|                                PRINTABLE STRING :
     |       |       |                                    'Administrator'
  833|     60|      1|                            SET :
  835|     58|      1|                              SEQUENCE :
  837|      9|      1|                                OBJECT IDENTIFIER : emailAddress
[1.2.840.113549.1.9.1]
  848|     45|      1|                                IA5 STRING :
     |       |       |                                    'Administrator@ssvsim-dom.extes'
     |       |       |                                    't.microsoft.com'
  895|    159|      2|                          SEQUENCE :
  898|     13|      1|                            SEQUENCE :
  900|      9|      1|                              OBJECT IDENTIFIER : rsaEncryption
[1.2.840.113549.1.1.1]
  911|      0|      1|                              NULL : ''
  913|    141|      2|                            BIT STRING UnusedBits:0 :
     |       |       |                          
  917|    137|      2|                              SEQUENCE :
  920|    129|      2|                                INTEGER :
     |       |       |                                    009B3B26A3CCF7D8F375B253D2AA95
     |       |       |                                    F2AE82AEEC1337732A455018689388
     |       |       |                                    75080011AD2F600B0B6FA11ABC91F1
     |       |       |                                    4020235E743D3FB250D071C606A338
     |       |       |                                    92F4657A30F55C4422ED2165C40E59
     |       |       |                                    884CF92EE84848D840DE6E75741955
     |       |       |                                    63C019546AFC59F81EC74F6D5543CE
     |       |       |                                    AFA5C972507AE1D773363B6406DD1E
     |       |       |                                    3AD11929E3F3839453
     |       |       |                                INTEGER : 65537
 1052|      3|      1|                          CONTEXT SPECIFIC (3) :
 1057|    371|      3|                            SEQUENCE :
 1061|    367|      3|                              SEQUENCE :
 1065|     14|      1|                                SEQUENCE :
 1067|      3|      1|                                  OBJECT IDENTIFIER : keyUsage [2.5.29.15]
 1072|      1|      1|                                  BOOLEAN :
     |       |       |                                      'ÿ'
 1075|      4|      1|                                  OCTET STRING :
 1077|      2|      1|                                    BIT STRING UnusedBits:4 :
     |       |       |                                        30
 1081|     19|      1|                                SEQUENCE :
 1083|      3|      1|                                  OBJECT IDENTIFIER : extKeyUsage [2.5.29.37]
 1088|     12|      1|                                  OCTET STRING :
 1090|     10|      1|                                    SEQUENCE :
```

*59 / 73*

```
 1092|    8|    1|                                        OBJECT IDENTIFIER : emailProtection
[1.3.6.1.5.5.7.3.4]
 1102|   29|    1|                                  SEQUENCE :
 1104|    3|    1|                                     OBJECT IDENTIFIER : subjectKeyIdentifier
[2.5.29.14]
 1109|   22|    1|                                     OCTET STRING :
 1111|   20|    1|                                        OCTET STRING :
     |     |     |                                           CAD24F57FA070174A0BB5C02594
     |     |     |                                           2E0C72AB912B3
 1133|   31|    1|                                  SEQUENCE :
 1135|    3|    1|                                     OBJECT IDENTIFIER : authorityKeyIdentifier
[2.5.29.35]
 1140|   24|    1|                                     OCTET STRING :
 1142|   22|    1|                                        SEQUENCE :
 1144|   20|    1|                                           CONTEXT SPECIFIC (0) :
     |     |     |                                              71DDD1A9BB0D7EE05F82FC9D
     |     |     |                                              C59D7A5E098E41EF
 1166|  107|    1|                                  SEQUENCE :
 1168|    3|    1|                                     OBJECT IDENTIFIER : cRLDistributionPoints
[2.5.29.31]
 1173|  100|    1|                                     OCTET STRING :
 1175|   98|    1|                                        SEQUENCE :
 1177|   96|    1|                                           SEQUENCE :
 1179|   94|    1|                                              CONTEXT SPECIFIC (0) :
 1181|   92|    1|                                                 CONTEXT SPECIFIC (0) :
 1183|   43|    1|                                                    CONTEXT SPECIFIC (6) :
     |     |     |                                                       'http://rsacerts'
     |     |     |                                                       'rv/CertEnroll/R'
     |     |     |                                                       'SACERTSRV.crl'
 1228|   45|    1|                                                    CONTEXT SPECIFIC (6) :
     |     |     |                                                       'file://\\rsacer'
     |     |     |                                                       'tsrv\CertEnroll'
     |     |     |                                                       '\RSACERTSRV.crl'
 1275|  154|    2|                                  SEQUENCE :
 1278|    8|    1|                                     OBJECT IDENTIFIER : authorityInfoAccess
[1.3.6.1.5.5.7.1.1]
 1288|  141|    2|                                     OCTET STRING :
 1291|  138|    2|                                        SEQUENCE :
 1294|   66|    1|                                           SEQUENCE :
 1296|    8|    1|                                              OBJECT IDENTIFIER : caIssuers
[1.3.6.1.5.5.7.48.2]
 1306|   54|    1|                                              CONTEXT SPECIFIC (6) :
     |     |     |                                                 'http://rsacertsrv/Cer'
     |     |     |                                                 'tEnroll/rsacertsrv_RS'
     |     |     |                                                 'ACERTSRV.crt'
 1362|   68|    1|                                           SEQUENCE :
 1364|    8|    1|                                              OBJECT IDENTIFIER : caIssuers
[1.3.6.1.5.5.7.48.2]
 1374|   56|    1|                                              CONTEXT SPECIFIC (6) :
     |     |     |                                                 'file://\\rsacertsrv\C'
     |     |     |                                                 'ertEnroll\rsacertsrv_'
     |     |     |                                                 'RSACERTSRV.crt'
 1432|   13|    1|                            SEQUENCE :
 1434|    9|    1|                               OBJECT IDENTIFIER : sha1withRSAEncryption
[1.2.840.113549.1.1.5]
 1445|    0|    1|                               NULL : ''
 1447|  129|    2|                            BIT STRING UnusedBits:0 :
     |     |     |                               8F409689A9F3125F18A1D52DC2C702533F993E3342
     |     |     |                               27967C07F682C27634D5DC65EA0A9BE66DF5E1BDF9
     |     |     |                               A2A780C4FC870BDF4B9D97D07F12B738B9B6636EBD
```

*60 / 73*

```
      |      |      |                          6B64E130BD85B100FA6AF676D5A318FBBFEEB03E2D
      |      |      |                          F86939B86F107753C89E1FAF76DDDA6B83F15AE950
      |      |      |                          5D754D3869B30DCA42F7C688AA935593E61ABC6F6A
      |      |      |                          F879
  1579|   649|    3|              SET :
  1583|   645|    3|                SEQUENCE :
  1587|     1|    1|                  INTEGER : 1
  1590|   106|    1|                  SEQUENCE :
  1592|    86|    1|                    SEQUENCE :
  1594|    60|    1|                      SET :
  1596|    58|    1|                        SEQUENCE :
  1598|     9|    1|                          OBJECT IDENTIFIER : emailAddress
[1.2.840.113549.1.9.1]
  1609|    45|    1|                          IA5 STRING :
      |      |      |                            'Administrator@ssvsim-dom.extes'
      |      |      |                            't.microsoft.com'
  1656|    22|    1|                      SET :
  1658|    20|    1|                        SEQUENCE :
  1660|     3|    1|                          OBJECT IDENTIFIER : commonName [2.5.4.3]
  1665|    13|    1|                          UTF8 STRING :
      |      |      |                            'Administrator'
  1680|    16|    1|                    INTEGER :
      |      |      |                      0274DE397E583BF0DB53CE0EECB024A0
  1698|     9|    1|                  SEQUENCE :
  1700|     5|    1|                    OBJECT IDENTIFIER : sha1 [1.3.14.3.2.26]
  1707|     0|    1|                    NULL : ''
  1709|   373|    3|                  CONTEXT SPECIFIC (0) :
  1713|    16|    1|                    SEQUENCE :
  1715|     9|    1|                      OBJECT IDENTIFIER :  [1.3.6.1.4.1.311.16.3]
  1726|     3|    1|                      SET :
  1728|     1|    1|                        INTEGER : 7
  1731|    24|    1|                    SEQUENCE :
  1733|     9|    1|                      OBJECT IDENTIFIER : contentType
[1.2.840.113549.1.9.3]
  1744|    11|    1|                      SET :
  1746|     9|    1|                        OBJECT IDENTIFIER : data
[1.2.840.113549.1.7.1]
  1757|    35|    1|                    SEQUENCE :
  1759|     9|    1|                      OBJECT IDENTIFIER : messageDigest
[1.2.840.113549.1.9.4]
  1770|    22|    1|                      SET :
  1772|    20|    1|                        OCTET STRING :
      |      |      |                          68DD2DD63498036679AD9081C69A564A4
      |      |      |                          A6A0E1A
  1794|    50|    1|                    SEQUENCE :
  1796|     9|    1|                      OBJECT IDENTIFIER : Outlook Express
[1.3.6.1.4.1.311.16.4]
  1807|    37|    1|                      SET :
  1809|    35|    1|                        SEQUENCE :
  1811|    21|    1|                          SEQUENCE :
  1813|    19|    1|                            SET :
  1815|    17|    1|                              SEQUENCE :
  1817|     3|    1|                                OBJECT IDENTIFIER : commonName
[2.5.4.3]
  1822|    10|    1|                                PRINTABLE STRING :
      |      |      |                                  'RSACERTSRV'
  1834|    10|    1|                          INTEGER :
      |      |      |                            7EC4FFE5000000001F5F
  1846|    52|    1|                        SEQUENCE :
```

```
1848|   11|     1|                                        OBJECT IDENTIFIER : [1.2.840.113549.1.9.16.2.11]
1861|   37|     1|                                        SET :
1863|   35|     1|                                          CONTEXT SPECIFIC (0) :
1865|   21|     1|                                            SEQUENCE :
1867|   19|     1|                                              SET :
1869|   17|     1|                                                SEQUENCE :
1871|    3|     1|                                                  OBJECT IDENTIFIER : commonName
[2.5.4.3]
1876|   10|     1|                                                  PRINTABLE STRING :
    |     |      |                                                    'RSACERTSRV'
1888|   10|     1|                                              INTEGER :
    |     |      |                                                7EC4FFE5000000001F5F
1900|  183|     2|                                  SEQUENCE :
1903|    9|     1|                                    OBJECT IDENTIFIER : sMIMECapabilities
[1.2.840.113549.1.9.15]
1914|  169|     2|                                    SET :
1917|  166|     2|                                      SEQUENCE :
1920|   11|     1|                                        SEQUENCE :
1922|    9|     1|                                          OBJECT IDENTIFIER :
[2.16.840.1.101.3.4.1.42]
1933|   11|     1|                                        SEQUENCE :
1935|    9|     1|                                          OBJECT IDENTIFIER :
[2.16.840.1.101.3.4.1.22]
1946|   10|     1|                                        SEQUENCE :
1948|    8|     1|                                          OBJECT IDENTIFIER : DES-EDE3-CBC
[1.2.840.113549.3.7]
1958|   11|     1|                                        SEQUENCE :
1960|    9|     1|                                          OBJECT IDENTIFIER :
[2.16.840.1.101.3.4.1.2]
1971|   14|     1|                                        SEQUENCE :
1973|    8|     1|                                          OBJECT IDENTIFIER : rc2CBC
[1.2.840.113549.3.2]
1983|    2|     1|                                          INTEGER : 128
1987|   13|     1|                                        SEQUENCE :
1989|    8|     1|                                          OBJECT IDENTIFIER : rc2CBC
[1.2.840.113549.3.2]
1999|    1|     1|                                          INTEGER : 64
2002|    7|     1|                                        SEQUENCE :
2004|    5|     1|                                          OBJECT IDENTIFIER : desCBC
[1.3.14.3.2.7]
2011|   13|     1|                                        SEQUENCE :
2013|    8|     1|                                          OBJECT IDENTIFIER : rc2CBC
[1.2.840.113549.3.2]
2023|    1|     1|                                          INTEGER : 40
2026|    7|     1|                                        SEQUENCE :
2028|    5|     1|                                          OBJECT IDENTIFIER : sha1 [1.3.14.3.2.26]
2035|   11|     1|                                        SEQUENCE :
2037|    9|     1|                                          OBJECT IDENTIFIER :
[2.16.840.1.101.3.4.2.3]
2048|   11|     1|                                        SEQUENCE :
2050|    9|     1|                                          OBJECT IDENTIFIER :
[2.16.840.1.101.3.4.2.2]
2061|   11|     1|                                        SEQUENCE :
2063|    9|     1|                                          OBJECT IDENTIFIER :
[2.16.840.1.101.3.4.2.1]
2074|   10|     1|                                        SEQUENCE :
2076|    8|     1|                                          OBJECT IDENTIFIER : md5
[1.2.840.113549.2.5]
2086|   13|     1|                                  SEQUENCE :
```

```
2088|     9|     1|                          OBJECT IDENTIFIER : rsaEncryption
 [1.2.840.113549.1.1.1]
2099|     0|     1|                          NULL : ''
2101|   128|     2|                      OCTET STRING :
    |      |      |                        0902E60039438E3D884700475000C6625CCD712FD0
    |      |      |                        125C0CCED7C541B6806FD8E609DD29E6E9703672B8
    |      |      |                        F976304652EF10F7ED07FB235E6D4DC2661AD0C3F3
    |      |      |                        F09CA561F4B5D014A200201634C3563C01EC5B60B2
    |      |      |                        332BF2F5C4707211B599FEA7C15FD58EFD8724FC94
    |      |      |                        D96F4A634D3EF21AB20FA2A2AE89225F41DD79776A
    |      |      |                        81DC
```

## 4.3  PidTagUserX509Certificate Property in Binary Format

The following example shows the structure of a sample **PidTagUserX509Certificate** property (section 2.2.4.66) value in binary format.

**Property Version Tag:**    0x0001

**Length of 8 bytes:**    0x0008

**Data:**    0x00000001

**KeyExchangeCertificate Tag:**    0x0003

**Length of 631 bytes:**    0x0277

**Data:**

3082026F308201D8A00302010202103F924830235200BC496308CD3BAEB76B300D06092A864886F
70D010105050003015311330110603550403130A5253414345525453525630301E170D303630333231
30353331329A170D31363033323130353431315A3015311330110603550403130A5253414143
4552545353655630819F300D06092A864886F70D010101050003818D003081890281810000B1AA1341E2
AD0E3D3E035B0F69FFD95CED15EBC9B85D74814F64E39D7F62E6CBB6627909ADE845ED578D56A
7710849BB3A084F0BD4AB5E9059448D365D667A5AC41B2B207344CF8EDB76FCCBDC0D80AF6A22
28D42CD38458542B80E21CC0780B9CAD962F6041820E02FBA4997E606DDFFD299A04D8C2C8B12
DCE47199D46935D0203010001A381BF3081BC300B0603551D0F040403020186300F0603551D130
101FF040530030101FF301D0603551D0E0416041471DDD1A9BB0D7EE05F82FC9DC59D7A5E098E4
1EF306B0603551D1F046430623060A05EA05C862B687474703A2F2F7273616363657274372762F43
657274456E726F6C6C2F525341434552525352562E63726C862D66696C653A2F2F5C5C72736163636
6572747372655C43657274456E726F6C6C5C52534143455252535262E63726C301006092B060104
018237150104030201003000D06092A864886F70D010105050003818100FECD3BBF82DA9AD38CE
804BC535AB39ED90F9F3A2BC73C79F43191CB250AFC4A77C43BA4D0D6DC108998551EC3127AC5
598C805F4B7DD76CAEDB431F8D37D53126F0FF9FE6D829B24F548B6CCA54361629D9B93EA9324A
A06C2EA3A39D9A1D29ED7EC07E9EE6E7C9D6805C1C7D07013A8B264216BBFCCF75330D6240A45
F3D6

**Encryption Type Tag:**    0x0006

**Length of 8 bytes:**    0x0008

**Data indicates S/MIME:**    0x00000001

**Defaults Tag:**    0x0020

**Length of 8 bytes:**    0x0008

**Data indicates an unspecified value:** 0x0000007

**AsymetricCapabilities:** 0x0002

**Length of 4 bytes:** 0x0004

# 5 Security

## 5.1 Security Considerations for Implementers

There are no special security considerations specific to this protocol. Note, however, that general security considerations pertaining to the underlying NSPI RPC-based transport, as described in [MS-NSPI] and [MS-OXNSPI], do apply to this protocol.

## 5.2 Index of Security Parameters

None.

# 6   Appendix A: 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 2003

- Microsoft® Exchange Server 2007

- Microsoft® Exchange Server 2010

- Microsoft® Exchange Server 2013 Preview

- Microsoft® Office Outlook® 2003

- Microsoft® Office Outlook® 2007

- Microsoft® Outlook® 2010

- Microsoft® Outlook® 2013 Preview

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

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

<1> Section 2.1: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the NSPI methods as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the NSPI methods as described in [MS-OXNSPI]. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory® global catalog server, in which case the server points the client to AD DS.

<2> Section 2.2.1.1: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the **PermanentEntryId** structure as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the **PermanentEntryId** structure as described in [MS-OXNSPI] section 2.3.8.3. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the **PermanentEntryId** structure as described in [MS-NSPI].

<3> Section 2.2.1.1: The OAB version 4 implementation in Exchange 2003, Exchange 2007, Exchange 2010, and Exchange 2013 Preview includes a **PidTagOfflineAddressBookTruncatedProperties** property ([MS-OXOAB] section 2.9.2.2) that is not included in objects on a NSPI server.

<4> Section 2.2.2: Retrieval of information about the address book hierarchy by using the OAB Public Folder Retrieval Protocol, as described in [MS-OXPFOAB], is not supported by Exchange 2013 Preview.

<5> Section 2.2.2.3: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the **NspiBind** method as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the **NspiBind** method as

described in [MS-OXNSPI] section 3.1.4.1.1. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the **NSPIBind** method as described in [MS-NSPI].

<6> Section 2.2.3: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the NSPI methods as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the NSPI methods as described in [MS-OXNSPI]. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the NSPI methods as described in [MS-NSPI].

<7> Section 2.2.3.2: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the **PermanentEntryID** and **EphemeralEntryID** structures as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the **PermanentEntryID** and **EphemeralEntryID** structures as described in [MS-OXNSPI] section 2.3.8.3 and section 2.3.8.2 respectively. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the **PermanentEntryID** and **EphemeralEntryID** structures as described in [MS-NSPI].

<8> Section 2.2.3.9: The **PidTagAddressBookPhoneticDisplayName** property (section 2.2.3.9) is not used in Exchange 2003 servers or Office Outlook 2003 clients.

<9> Section 2.2.3.11: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the display type values as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the display type values as described in [MS-OXNSPI] section 2.2.3. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the display type values as described in [MS-NSPI].

<10> Section 2.2.3.12: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the display type values as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the display type values as described in [MS-OXNSPI] section 2.2.3. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the display type values as described in [MS-NSPI].

<11> Section 2.2.3.26: The **PidTagAddressBookSenderHintTranslations** property (section 2.2.3.26) is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<12> Section 2.2.3.27: The **PidTagAddressBookDeliveryContentLength** property (section 2.2.3.27) is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<13> Section 2.2.3.28: The **PidTagAddressBookModerationEnabled** property (section 2.2.3.28) is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<14> Section 2.2.3.29: The **PidTagAddressBookDistributionListMemberCount** property (section 2.2.3.29) is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<15> Section 2.2.3.30: The **PidTagAddressBookDistributionListExternalMemberCount** property (section 2.2.3.30) is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<16> Section 2.2.3.35: The **PidTagAddressBookDisplayTypeExtended** property (section 2.2.3.35) is not used in Exchange 2003 servers or Office Outlook 2003 clients.

<17> Section 2.2.4.14: The **PidTagAddressBookPhoneticGivenName** property (section 2.2.4.14) is not used in Exchange 2003 servers or Office Outlook 2003 clients.

<18> Section 2.2.4.15: The **PidTagAddressBookPhoneticSurname** property (section 2.2.4.15) is not used in Exchange 2003 servers or Office Outlook 2003 clients.

<19> Section 2.2.4.16: The **PidTagAddressBookPhoneticCompanyName** property (section 2.2.4.16) is not used by Exchange 2003 servers or Office Outlook 2003 clients.

<20> Section 2.2.4.17: The **PidTagAddressBookPhoneticDepartmentName** property (section 2.2.4.17) is not used by Exchange 2003 servers or Office Outlook 2003 clients.

<21> Section 2.2.4.66.4: The **PidTagAddressBookPhoneticDisplayName** property (section 2.2.3.9) is not used in Exchange 2003 or Office Outlook 2003 servers.

<22> Section 2.2.4.66.5: The **PidTagAddressBookPhoneticDisplayName** property (section 2.2.3.9) is not used in Exchange 2003 or Office Outlook 2003 servers.

<23> Section 2.2.4.66.7: The **PidTagAddressBookPhoneticDisplayName** property (section 2.2.3.9) is not used in Exchange 2003 or Office Outlook 2003 servers.

<24> Section 2.2.4.66.9: The **PidTagAddressBookPhoneticDisplayName** property (section 2.2.3.9) is not used in Exchange 2003 or Office Outlook 2003 servers.

<25> Section 2.2.4.81: The **PidTagAddressBookOrganizationalUnitRootDistinguishedName** property (section 2.2.4.81) is not used by Office Outlook 2003 clients.

<26> Section 2.2.4.82: The **PidTagThumbnailPhoto** property (section 2.2.4.82) is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<27> Section 2.2.4.83: The **PidTagSpokenName** property (section 2.2.4.83) is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<28> Section 2.2.4.84: The **PidTagAddressBookAuthorizedSenders** property (section 2.2.4.84) is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<29> Section 2.2.4.85: The **PidTagAddressBookUnauthorizedSenders** property (section 2.2.4.85) is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<30> Section 2.2.4.86: The **PidTagAddressBookDistributionListMemberSubmitAccepted** property (section 2.2.4.86) is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<31> Section 2.2.4.87: The **PidTagAddressBookDistributionListMemberSubmitRejected** property (section 2.2.4.87) is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<32> Section 2.2.5: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the tables as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the tables as described in [MS-OXNSPI] section 3.1.4.4.2.2. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the tables as described in [MS-NSPI].

<33> Section 2.2.5: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the **NspiGetMatches** and **NspiModLinkAtt** methods as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the **NspiGetMatches** and **NspiModLinkAtt** methods, as described in [MS-OXNSPI] section 3.1.4.1.10 and section 3.1.4.1.15 respectively. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory® global catalog server, in which case the server points the client to AD DS and implements the **NspiGetMatches** and **NspiModLinkAtt** methods as described in [MS-NSPI].

<34> Section 2.2.7: The **PidTagAddressBookOrganizationalUnitRootDistinguishedName** property (section 2.2.4.81) is not used by Office Outlook 2003 clients.

<35> Section 2.2.7.2: Office Outlook 2003 clients do not recognize departmental groups. If the department hierarchy is represented by using departmental groups, these clients run as if the **PidTagAddressBookHierarchicalRootDepartment** property (section 2.2.7.2) is not present, and there is no hierarchical representation of the departments.

<36> Section 2.2.8: If a server represents a department hierarchy by using both Department objects and departmental groups, Office Outlook 2003 only makes use of the hierarchy that is represented by the Department objects, and Office Outlook 2007, Outlook 2010, and Outlook 2013 Preview only make use of the hierarchy that is represented by the departmental groups. Note that a single hierarchy cannot be represented by a combination of the two schemas.

<37> Section 2.2.10.1: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the **NspiGetMatches** methods as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the NspiGetMatches methods as described in [MS-OXNSPI] section 3.1.4.1.10. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the **NspiGetMatches** method as described in [MS-NSPI].

<38> Section 2.2.10.1: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements ANR as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements ANR as described in [MS-OXNSPI] section 3.1.4.7. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the ANR as described in [MS-NSPI].

<39> Section 2.2.10.1: Office Outlook 2003, Office Outlook 2007, Outlook 2010, and Outlook 2013 Preview perform a lookup of SMTP addresses by constructing a specific target string for ambiguous name resolution (ANR) that is understood by Exchange 2003, Exchange 2007, Exchange 2010, and Exchange 2013 Preview servers. When the target string starts with "=SMTP: " and is followed by a valid SMTP address, the NSPI server returns exactly one match, if any are found. The match will be an address book entry that has the target string as a valid SMTP address. Such a target string can be specified as a **PidTagAnr** property restriction to the **NspiGetMatches** method, as specified in section 2.2.3.26. Such a target string is also understood by the *paStr* parameter to the **NspiResolveNames** method, as specified in [MS-OXNSPI] section 3.1.4.1.16, or the *paWStr* parameter to the **NspiResolveNamesW** method, as specified in [MS-OXNSPI] section 3.1.4.1.17.

<40> Section 3.1.3.1: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the **NspiBind** method as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the **NspiBind** method as described in [MS-OXNSPI] section 3.1.4.1.1. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the **NspiBind** method as described in [MS-NSPI].

*Release: July 16, 2012*

<41> Section 3.1.3.2: Retrieval of address lists by using the OAB Public Folder Retrieval Protocol, as described in [MS-OXPFOAB], is not supported by Exchange 2013 Preview.

<42> Section 3.1.4.1: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the **NspiGetSpecialTable** method as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the **NspiGetSpecialTable** method as described in [MS-OXNSPI] section 3.1.4.1.3. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the **NspiGetSpecialTable** method as described in [MS-NSPI].

<43> Section 3.1.4.1: Retrieval of hierarchy tables by using the OAB Public Folder Retrieval Protocol, as described in [MS-OXPFOAB], is not supported by Exchange 2013 Preview.

<44> Section 3.1.4.2: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the **STAT** structure as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the **STAT** structure as described in [MS-OXNSPI] section 2.3.7. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the **STAT** structure as described in [MS-NSPI].

<45> Section 3.1.4.2: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the NSPI method as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the NSPI methods as described in [MS-OXNSPI] section 3.1.4.1.4. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the NSPI methods as described in [MS-NSPI].

<46> Section 3.1.4.3: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the NSPI methods as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the NSPI methods as described in [MS-OXNSPI]. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the NSPI methods as described in [MS-NSPI].

<47> Section 3.1.4.3: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the **PermanentEntryID** structure as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the **PermanentEntryID** structure as described in [MS-OXNSPI] section 2.3.8.3. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the **PermanentEntryID** structure as described in [MS-NSPI].

<48> Section 3.1.4.4: Exchange 2003 and Exchange 2007 point the client to AD DS, which implements the NSPI methods as described in [MS-NSPI]. Exchange 2010 and Exchange 2013 Preview point the client to the Exchange NSPI server, which implements the NSPI methods as described in [MS-OXNSPI]. The only exception is when Exchange 2010 or Exchange 2013 Preview are installed on an Active Directory global catalog server, in which case the server points the client to AD DS and implements the NSPI methods as described in [MS-NSPI].

# 7 Change Tracking

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

# 8   Index

A

Abstract data model
  client 46
  server 50
Applicability 13

C

Capability negotiation 13
Change tracking 71
Client
  abstract data model 46
  higher-layer triggered events 47
  initialization 47
  message processing 49
  other local events 50
  sequencing rules 49
  timer events 50
  timers 47

D

Data model - abstract
  client 46
  server 50
Definitions message 15

F

Fields - vendor-extensible 13

G

Glossary 9

H

Higher-layer triggered events
  client 47
  server 51

I

Implementer - security considerations 65
Index of security parameters 65
Informative references 12
Initialization
  client 47
  server 51
Introduction 9

M

Message processing
  client 49
  server 51
Messages
  Definitions 15

Named Properties 45
Properties that Apply to All Address Book Objects
  18
Properties that Apply to Containers in the
  Address Book Hierarchy Table 16
Properties That Apply to Department Objects 44
Properties That Apply to Organization Objects 43
Properties That Reference Other Address Book
  Objects 41
transport 14

N

Named Properties message 45
Normative references 10

O

Other local events
  client 50
  server 51
Overview (synopsis) 12

P

Parameters - security index 65
Preconditions 12
Prerequisites 12
Product behavior 66
Properties that Apply to All Address Book Objects
  message 18
Properties that Apply to Containers in the Address
  Book Hierarchy Table message 16
Properties That Apply to Department Objects
  message 44
Properties That Apply to Organization Objects
  message 43
Properties That Reference Other Address Book
  Objects message 41

R

References 10
  informative 12
  normative 10
Relationship to other protocols 12

S

Security
  implementer considerations 65
  parameter index 65
Sequencing rules
  client 49
  server 51
Server
  abstract data model 50
  higher-layer triggered events 51
  initialization 51