[MS-OXOABK]:
Address Book Object Protocol Specification

Intellectual Property Rights Notice for Open Specifications Documentation

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	4.0.0	No change	No changes to the meaning, language, or formatting of the technical content.
11/03/2010	4.1	Minor	Clarified the meaning of the technical content.
03/18/2011	5.0	Major	Significantly changed the technical content.

Table of Contents

1 Introduction

An **address book** is a repository of information about **Address Book objects**, many of which have information that can be retrieved by messaging clients to display to messaging users, or to enable users to address e-mail messages to other e-mail-enabled users and entities that are represented by these objects. The objects in the address book are organized in a way that makes it possible for users to locate and look up information about these objects.

The client retrieves data about a number of different Address Book objects from one of two possible data sources. The client obtains data from the **name service provider interface (NSPI)** server by using the protocol specified in [MS-NSPI]. Alternatively, the client gets data from an **offline address book (OAB)**, as specified in [MS-OXOAB]. These include properties on **mail users**, **distribution lists**, **resources**, **address book containers**, and the **address book hierarchy table**. The client chooses the data source based on whatever source is available, or based on user settings.

This specification defines the following:

- The properties of various Address Book objects.

- How the properties of Address Book objects interrelate with one another.

Sections 1.8, 2, and 3 of this specification are normative and contain RFC 2119 language. Section 1.5 and 1.9 are also normative but cannot contain RFC 2119 language. 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**
> **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)**
> **resource**
> **Unicode**

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

> **address book**
> **address book container**
> **Address Book object**
> **address list**
> **address type**
> **alias**
> **binary large object (BLOB)**
> **contact**

departmental group
display template
distribution list
entry ID
File Transfer Protocol (FTP)
Global Address List (GAL)
hierarchy table
locale
mail tip
mail user
mailbox
meeting request
messaging object
Multipurpose Internet Mail Extensions (MIME)
named property
offline address book (OAB)
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:

address book hierarchy table: A collection of address book containers arranged in a hierarchy.

Department object: An Address Book object that describes a department within an organization.

Minimal Entry ID: A property of an Address Book object that can be used to uniquely identify the object.

Organization object: An Address Book object that describes an entire organization.

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

1.2   References

1.2.1   Normative References

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

http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624, as an additional source.

[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", June 2008.

[MS-OXABREF] Microsoft Corporation, "Address Book Name Service Provider Interface (NSPI) Referral Protocol Specification", April 2008.

[MS-OXCDATA] Microsoft Corporation, "Data Structures", April 2008.

[MS-OXCMAIL] Microsoft Corporation, "RFC2822 and MIME to E-Mail Object Conversion Protocol Specification", June 2008.

[MS-OXOAB] Microsoft Corporation, "Offline Address Book (OAB) File Format and Schema", June 2008.

[MS-OXOABKT] Microsoft Corporation, "Address Book User Interface Templates Protocol Specification", April 2008.

[MS-OXPFOAB] Microsoft Corporation, "Offline Address Book (OAB) Public Folder Retrieval Protocol Specification", June 2008.

[MS-OXWOAB] Microsoft Corporation, "Offline Address Book (OAB) Retrieval File Format", June 2008.

[MS-RPCE] Microsoft Corporation, "Remote Procedure Call Protocol Extensions", January 2007.

[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.ietf.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

[RFC3851] Ramsdell, B., Ed., "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.1 Message Specification", RFC 3851, July 2004, http://www.rfc-editor.org/rfc/rfc3851.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.ietf.org/rfc/rfc5234.txt

1.2.2 Informative References

[MS-GLOS] Microsoft Corporation, "Windows Protocols Master Glossary", March 2007.

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

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 the NSPI protocol, as specified in [MS-NSPI]. 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 specified in [MS-OXOAB].

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 specified in [MS-NSPI], and an understanding of the Offline Address Book (OAB) Format and Schema Protocol, as specified 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 specified in [MS-OXABREF]) and an understanding of how to obtain an OAB by using either the Offline Address Book Web Retrieval protocol (as specified in [MS-OXWOAB]) or the Offline Address Book Public Folder Retrieval Protocol (as specified in [MS-OXPFOAB]). The Address Book User Interface Templates Protocol, as specified in [MS-OXOABKT], is needed to render information about an Address Book object to a messaging user, which relies on an understanding of this Address Book Object 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, as specified in [MS-NSPI], or that the messaging client has access to an OAB, as specified in [MS-OXOAB].

1.6 Applicability Statement

This Address Book Object 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

This protocol uses the Name Service Provider Interface (NSPI) Protocol, as specified in [MS-NSPI], as the underlying transport protocol when handling address books online by using an NSPI server, and it uses the Offline Address Book (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 the Name Service Provider Interface (NSPI) Protocol, as specified in [MS-NSPI]. 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 Offline Address Book (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 Offline Address Book (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** that are listed in this document, a request to an NSPI server of that property with type **PtypString8** or **PtypString** is permitted. If a conversion is required, the string will be converted to the type requested by the client, as specified in [MS-NSPI].

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 Offline Address Book (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

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)** 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** (section 2.2.3.14) property. The DN(1) is also embedded in the distinguished name field of the **Permanent Entry ID**, as specified in [MS-NSPI]. 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" / "|"
```

In general, the DNs(1) for all Address Book objects MUST follow the *dn* format. The DNs(1) for other types of objects MUST follow more strict *dn* formats, depending on their type, as shown in the following table.

Object type	DN format	Notes
Address book container	*addresslist-dn*	
Global Address List container	*gal-addrlist-dn*	

Object type	DN format	Notes
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)** in the object-rdn is the name of the mailbox server.
Room container reference	x500-dn with no container-rdn	The RDN 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<1> be a subset of the properties in the NSPI server, MAY<2> include additional properties, and SHOULD<3><4> have the same value when present on both data sources.

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

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 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], and 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.

When using an NSPI server, messaging clients call the **NspiGetSpecialTable** method, as specified in [MS-NSPI], to obtain the address book hierarchy table. The hierarchy table is a set of rows, each of which describes one address book container. The NSPI server MUST return the following properties for each container in the hierarchy, in the order listed:

- **PidTagEntryId** (section 2.2.3.2)

- **PidTagContainerFlags** (section 2.2.2.1)

- **PidTagDepth** (section 2.2.2.2)

- **PidTagAddressBookContainerId** (section 2.2.2.3)

- **PidTagDisplayName** (section 2.2.3.1)

- **PidTagAddressBookIsMaster** (section 2.2.3.4)

- **PidTagAddressBookParentEntryId** (section 2.2.2.5) (optional, and MUST be the seventh column if it is included)

For every row returned, all of these properties except **PidTagAddressBookParentEntryId** (section 2.2.2.5) MUST be present, and each of them MUST have a value prescribed under its definition.

In addition, the **PidTagEntryId** property (section 2.2.3.2) MUST be in the form of a Permanent Entry ID, as specified in [MS-NSPI], with its PidTag**DisplayType** property having the value DT_CONTAINER, as specified in [MS-NSPI], and its DN following the *addresslist-dn* format specification, as specified in section 2.2.1.1. When the object is the Global Address List (GAL) container, its DN MUST follow the *gal-addrlist-dn* format specification.

2.2.2.1 PidTagContainerFlags

The **PidTagContainerFlags** property is a **PtypInteger32** value that contains a bitmask of **flags** that describe capabilities of an address book container.

The flags listed in the following table are defined for the **PidTagContainerFlags** bitmask. The **PidTagContainerFlags** property (section 2.2.2.1) 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 Offline Address Book (OAB) Retrieval File Format, as specified in [MS-OXWOAB], and the Offline Address Book (OAB) Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB].

2.2.2.2 PidTagDepth

The **PidTagDepth** property is a **PtypInteger32** value that represents the relative level of depth of a container in a hierarchy table. Objects in the hierarchy table that share the same **PidTagDepth** value 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 Offline Address Book (OAB) Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB].

2.2.2.3 PidTagAddressBookContainerId

The **PidTagAddressBookContainerId** property is a **PtypInteger32** value, like the **PidTagEntryId** property (section 2.2.3.2), and it 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 **NspiResolveNamesW**) 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 *pServerGuid*, as specified in [MS-NSPI].

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 Offline Address Book (OAB) Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB].

2.2.2.4 PidTagAddressBookIsMaster

The **PidTagAddressBookIsMaster** property is a **PtypBoolean** value that is "TRUE" if it is possible to create Address Book objects in that container, and "FALSE" 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 Offline Address Book (OAB) Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB].

2.2.2.5 PidTagAddressBookParentEntryId

The **PidTagAddressBookParentEntryId** property is a **PtypBinary** value that 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 Offline Address Book (OAB) Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB].

2.2.3 Properties that Apply to All Address Book Objects

Address Book objects are listed in an offline address book (OAB) by the Offline Address Book (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 calls, as specified in [MS-NSPI]. 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 MAY<5> have additional properties and operations, such as the ability to determine the distribution list membership. Mail user objects are further broken down into subtypes, such as rooms, equipment, messaging forums, or other types.

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.

All Address Book objects that are obtained from an OAB or an NSPI server MUST define values for the following properties:

- **PidTagDisplayName (**section 2.2.3.1)

- **PidTagEmailAddress** (section 2.2.3.14)

- **PidTagObjectType** (section 2.2.3.10)

- **PidTagDisplayType** (section 2.2.3.11)

In addition, all Address Book objects that are obtained from an NSPI server MUST define values for the following property:

- **PidTag7BitDisplayName** (section 2.2.3.7)

In addition, all Address Book objects that are obtained from an NSPI server MUST define values for the following properties, which MUST NOT be defined for any objects in an OAB:

- **PidTagEntryId** (section 2.2.3.2)

- **PidTagTemplateid** (section 2.2.3.3)

- **PidTagRecordKey** (section 2.2.3.4)

- **PidTagSearchKey** (section 2.2.3.5)

- **PidTagInstanceKey** (section 2.2.3.5)

- **PidTagAddressBookContainerId** (section 2.2.2.3)

- **PidTagInitialDetailsPane** (section 2.2.3.33)

- **PidTagTransmittableDisplayName** (section 2.2.3.8)

- **PidTagAddressType** (section 2.2.3.13)

- **PidTagAddressBookObjectDistinguishedName** (section 2.2.3.15)

In addition, an OAB MUST NOT define values for the following properties:

- **PidTagDepth** (section 2.2.2.2)

- **PidTagContainerFlags** (section 2.2.2.1)

An NSPI server MUST define values for the following properties for distribution lists and define no values for other object types:

- **PidTagContainerContents** (section 2.2.6.3)

- **PidTagContainerFlags** (section 2.2.2.1)

An OAB does not contain the following properties, but values MUST be generated for them when this protocol is being implemented:

- **PidTag7BitDisplayName** (section 2.2.3.7)

- **PidTagDepth** (section 2.2.2.2)

- **PidTagContainerFlags** (section 2.2.2.1)

If either an NSPI server or OAB does not conform to the preceding rules, and to the descriptions of the properties in the following sections, client behavior is undefined.

2.2.3.1   PidTagDisplayName

The **PidTagDisplayName** property is a **PtypString** value that represents a displayable form of an Address Book object. When the Address Book object is a mail user, the **PidTagDisplayName**

property is the name of the mail user, usually the mail user's full name. When the object is a distribution list, **PidTagDisplayName** 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, **PidTagDisplayName** 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 **PidTagDisplayName** value for each row in the hierarchy table represents the name of the address book container of that row.

2.2.3.2   PidTagEntryId

The **PidTagEntryId** property is a **PtypBinary** value that 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 is a **PtypBinary** value that identifies an Address Book object on an NSPI server and represents either the Permanent Entry ID or the Ephemeral Entry ID, as specified in [MS-NSPI]. 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 objectare specified in [MS-NSPI]. When the object is in Permanent Entry ID format, its DN MUST match the value for **PidTagEmailAddress** (section 2.2.3.14) and MUST follow the *dn* format specification that is particular to the type of object, as described in section 2.2.1.1.

The Offline Address Book (OAB) Format and Schema Protocol specification ([MS-OXOAB]) does not include **PidTagEntryId** values 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

The **PidTagTemplateid** property is a **PtypBinary** value that 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 MUST match the value for **PidTagEmailAddress** (section 2.2.3.14), and its DN MUST follow the *dn* format specification 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

The **PidTagRecordKey** property is a **PtypBinary** value that 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

The **PidTagSearchKey** property is a **PtypBinary** value formed by concatenating the ASCII string "EX: " followed by the DN 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

The **PidTagInstanceKey** property is a **PtypBinary** value, like the **PidTagEntryId** property (section 2.2.3.2), and it 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 PidTag7BitDisplayName

The **PidTag7BitDisplayName** property is a **PtypString** value that 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

The **PidTagTransmittableDisplayName** property is a **PtypString** value that 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

The **PidTagAddressBookPhoneticDisplayName** property is a **PtypString** value that is the phonetic representation of the **PidTagDisplayName** property (section 2.2.3.1).<6>

2.2.3.10 PidTagObjectType

The **PidTagObjectType** property is a **PtypInteger32** value containing 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

The **PidTagDisplayType** property is a **PtypInteger32** value containing 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.

PidTagDisplayType MUST have one of the following display type values, as specified in [MS-NSPI], according to the object's type: **DT_MAILUSER**, **DT_DISTLIST**, **DT_FORUM**, **DT_AGENT**, **DT_ORGANIZATION**, **DT_PRIVATE_DISTLIST**, or **DT_REMOTE_MAILUSER**.

2.2.3.12 PidTagDisplayTypeEx

The **PidTagDisplayTypeEx** property is a **PtypInteger32** value containing a value that indicates how to display an Address Book object in a table or as a **recipient** 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 **PidTagDisplayType** (section 2.2.3.11). In addition to the display information contained in **PidTagDisplayType**, this property distinguishes among additional object types. When the object comes from a remote server, **PidTagDisplayType** 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], 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	0x00000007	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	0x00000008	Equipment. It is possible for messaging clients to send meeting requests to the specified Address Book object to reserve the equipment.
DT_SEC_DISTLIST	0x00000009	A distribution list used as a security group on the server.

2.2.3.13   PidTagAddressType

The **PidTagAddressType** property is a **PtypString** value that 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

The **PidTagEmailAddress** property is a **PtypString** value that contains an Address Book object's e-mail address, expressed in X500 format, using the *dn* format specification 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 of the Permanent Entry ID for the object if the object is present on an NSPI server. Its DN MUST follow the *dn* format specification particular to the type of object, as specified in section 2.2.1.1.

2.2.3.15   PidTagAddressBookObjectDistinguishedName

The **PidTagAddressBookObjectDistinguishedName** property is a **PtypString** value that contains the distinguished name (DN) of an Address Book object in DN format, as specified in section 2.2.1.1. If present, its DN MUST follow the *dn* format specification 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

The **PidTagCreationTime** property is a **PtypTime** value that contains the creation date and time for an Address Book object in **Coordinated Universal Time (UTC)**.

2.2.3.17   PidTagLastModificationTime

The **PidTagLastModificationTime** property is a **PtypTime** value that contains the date and time that an Address Book object was last modified in Coordinated Universal Time (UTC).

2.2.3.18   PidTagSendRichInfo

The **PidTagSendRichInfo** property is a **PtypBoolean** value that contains "TRUE" 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].

2.2.3.19   PidTagSendInternetEncoding

The **PidTagSendInternetEncoding** property is a **PtypInteger32** value that 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

The PidTagAccount property is a **PtypString** value that contains an Address Book object's **alias**, which is an alternative name by which the object can be identified.

2.2.3.21   PidTagSmtpAddress

The **PidTagSmtpAddress** property is a **PtypString** value that contains an Address Book object's **Simple Mail Transfer Protocol (SMTP)** address.

2.2.3.22   PidTagAddressBookTargetAddress

The **PidTagAddressBookTargetAddress** property is a **PtypString** value that 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

The **PidTagAddressBookProxyAddresses** property is a **PtypMultipleString** value that 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

The **PidTagAddressBookSeniorityIndex** property is a **PtypInteger32** value containing 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

The **PidTagAddressBookObjectGuid** property is a **PtypBinary** value containing a GUID that uniquely identifies an Address Book object.

2.2.3.26   PidTagAddressBookSenderHintTranslations

The **PidTagAddressBookSenderHintTranslations** property<7> is a **PtypMultipleString** value that 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

The **PidTagAddressBookDeliveryContentLength** property<8> is a **PtypInteger32** value that specifies the maximum size of a message that a recipient can receive.

2.2.3.28 PidTagAddressBookModerationEnabled

The **PidTagAddressBookModerationEnabled** property<9> is a **PtypBoolean** value that contains "TRUE if moderation is enabled for the mail user or distribution list; otherwise, this property contains "FALSE".

2.2.3.29 PidTagAddressBookDistributionListMemberCount

The **PidTagAddressBookDistributionListMemberCount** property<10> is a **PtypInteger32** value that contains the total number of recipients 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

The **PidTagAddressBookDistributionListExternalMemberCount** property<11> is a **PtypInteger32** value that contains the number of external recipients in a specified distribution list.

2.2.3.31 PidTagComment

The **PidTagComment** property is a **PtypString** value that contains a comment about the purpose or content of an Address Book object.

2.2.3.32 PidTagMappingSignature

The **PidTagMappingSignature** property is a **PtypBinary** value that 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

The **PidTagInitialDetailsPane** property is a **PtypInteger32** value that 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 that are defined and populated by the organization that is to be added to display templates.

2.2.3.35 PidTagAddressBookDisplayTypeExtended

This property is limited to **Recipient objects**. The **PidTagAddressBookDisplayTypeExtended** property SHOULD be present on objects on an NSPI server or an offline address book (OAB).<12>

2.2.4 Properties that Apply to Mail User Objects

2.2.4.1 PidTagSurname

The **PidTagSurname** property is a **PtypString** value that contains a mail user's family name.

2.2.4.2 PidTagGivenName

The **PidTagGivenName** property is a **PtypString** value that contains a mail user's given name.

2.2.4.3 PidTagNickname

The **PidTagNickname** property is a **PtypString** value that contains a mail user's nickname.

2.2.4.4 PidTagDisplayNamePrefix

The **PidTagDisplayNamePrefix** property is a **PtypString** value that contains a mail user's honorific title.

2.2.4.5 PidTagInitials

The **PidTagInitials** property is a **PtypString** value that contains the initials for parts of a mail user's full name.

2.2.4.6 PidTagGeneration

The **PidTagGeneration** property is a **PtypString** value that contains a generational abbreviation that follows a mail user's full name.

2.2.4.7 PidTagTitle

The **PidTagTitle** property is a **PtypString** value that contains a mail user's job title.

2.2.4.8 PidTagOfficeLocation

The **PidTagOfficeLocation** property is a **PtypString** value that contains a mail user's office location.

2.2.4.9 PidTagDepartmentName

The **PidTagDepartmentName** property is a **PtypString** value that contains a name for the department in which a mail user works.

2.2.4.10 PidTagCompanyName

The **PidTagCompanyName** property is a **PtypString** value that contains a mail user's company name.

2.2.4.11 PidTagAssistant

The **PidTagAssistant** property is a **PtypString** value that contains the name of a mail user's administrative assistant.

2.2.4.12   PidTagManagerName

The **PidTagManagerName** property is a **PtypString** value that contains the name of a mail user's manager.

2.2.4.13   PidTagAddressBookManagerDistinguishedName

The **PidTagAddressBookManagerDistinguishedName** property is a **PtypString** value that contains the distinguished name (DN) of a mail user's manager.

2.2.4.14   PidTagAddressBookPhoneticGivenName

The **PidTagAddressBookPhoneticGivenName** property is a **PtypString** value tht contains the phonetic representation of the **PidTagGivenName** property (section 2.2.4.2).<13>

2.2.4.15   PidTagAddressBookPhoneticSurname

The **PidTagAddressBookPhoneticSurname** property is a **PtypString** value that contains the phonetic representation of the **PidTagSurname** property (section 2.2.4.1).<14>

2.2.4.16   PidTagAddressBookPhoneticCompanyName

The **PidTagAddressBookPhoneticCompanyName** property is a **PtypString** value that contaiins the phonetic representation of the **PidTagCompanyName** property (section 2.2.4.10).<15>

2.2.4.17   PidTagAddressBookPhoneticDepartmentName

The **PidTagAddressBookPhoneticDepartmentName** property is a **PtypString** value that contains the phonetic representation of the **PidTagDepartmentName** property (section 2.2.4.9).<16>

2.2.4.18   PidTagPostalAddress

The **PidTagPostalAddress** property is a **PtypString** value that contains a mail user's postal address.

2.2.4.19   PidTagStreetAddress

The **PidTagStreetAddress** property is a **PtypString** value that contains a mail user's street address.

2.2.4.20   PidTagPostOfficeBox

The **PidTagPostOfficeBox** property is a **PtypString** value that contains the number or identifier of a mail user's post office box.

2.2.4.21   PidTagLocality

The **PidTagLocality** property is a **PtypString** value that contains the name of a mail user's locality, such as the town or city.

2.2.4.22   PidTagStateOrProvince

The **PidTagStateOrProvince** property is a **PtypString** value that contains the name of a mail user's state or province.

2.2.4.23   PidTagPostalCode

The **PidTagPostalCode** property is a **PtypString** value that contains the postal code for a mail user's postal address.

2.2.4.24   PidTagCountry

The **PidTagCountry** property is a **PtypString** value that contains the name of a mail user's country/region.

2.2.4.25   PidTagHomeAddressStreet

The **PidTagHomeAddressStreet** property is a **PtypString** value that contains a mail user's home street address.

2.2.4.26   PidTagHomeAddressPostOfficeBox

The **PidTagHomeAddressPostOfficeBox** property is a **PtypString** value that contains the number or identifier of a mail user's home post office box.

2.2.4.27   PidTagHomeAddressCity

The **PidTagHomeAddressCity** property is a **PtypString** value that contains the name of a mail user's home locality, such as the town or city.

2.2.4.28   PidTagHomeAddressStateOrProvince

The **PidTagHomeAddressStateOrProvince** property is a **PtypString** value that contains the name of a mail user's home state or province.

2.2.4.29   PidTagHomeAddressPostalCode

The **PidTagHomeAddressPostalCode** property is a **PtypString** value that contains the postal code for a mail user's home postal address.

2.2.4.30   PidTagHomeAddressCountry

The **PidTagHomeAddressCountry** property is a **PtypString** value that contains the name of a mail user's home country/region.

2.2.4.31   PidTagOtherAddressStreet

The **PidTagOtherAddressStreet** property is a **PtypString** value that contains a mail user's other street address.

2.2.4.32   PidTagOtherAddressPostOfficeBox

The **PidTagOtherAddressPostOfficeBox** property is a **PtypString** value that contains the number or identifier of a mail user's other post office box.

2.2.4.33 PidTagOtherAddressCity

The **PidTagOtherAddressCity** property is a **PtypString** value that contains the name of a mail user's other locality, such as the town or city.

2.2.4.34 PidTagOtherAddressStateOrProvince

The **PidTagOtherAddressStateOrProvince** property is a **PtypString** value that contains the name of a mail user's other state or province.

2.2.4.35 PidTagOtherAddressPostalCode

The **PidTagOtherAddressPostalCode** property is a **PtypString** value that contains the postal code for a mail user's other postal address.

2.2.4.36 PidTagOtherAddressCountry

The **PidTagOtherAddressCountry** property is a **PtypString** value that contains the name of a mail user's other country/region.

2.2.4.37 PidTagPrimaryTelephoneNumber

The **PidTagPrimaryTelephoneNumber** property is a **PtypString** value that contains a mail user's primary telephone number.

2.2.4.38 PidTagBusinessTelephoneNumber

The **PidTagBusinessTelephoneNumber** property is a **PtypString** value that contains the primary telephone number of a mail user's place of business.

2.2.4.39 PidTagHomeTelephoneNumber

The **PidTagHomeTelephoneNumber** property is a **PtypString** value that contains the primary telephone number of a mail user's home.

2.2.4.40 PidTagBusiness2TelephoneNumber

The **PidTagBusiness2TelephoneNumber** property is a **PtypString** value that contains a secondary telephone number at a mail user's place of business.

2.2.4.41 PidTagBusiness2TelephoneNumbers

The **PidTagBusiness2TelephoneNumbers** property is a **PtypMultipleString** value that contains secondary telephone numbers at a mail user's place of business.

2.2.4.42 PidTagHome2TelephoneNumber

The **PidTagHome2TelephoneNumber** property is a **PtypString** value that contains a secondary telephone number at a mail user's home.

2.2.4.43 PidTagHome2TelephoneNumbers

The **PidTagHome2TelephoneNumbers** property is a **PtypMultipleString** value that contains secondary telephone numbers at a mail user's home.

2.2.4.44 PidTagCallbackTelephoneNumber

The **PidTagCallbackTelephoneNumber** property is a **PtypString** value that contains a callback telephone number for a mail user.

2.2.4.45 PidTagMobileTelephoneNumber

The **PidTagMobileTelephoneNumber** property is a **PtypString** value that contains a mail user's mobile telephone number.

2.2.4.46 PidTagRadioTelephoneNumber

The **PidTagRadioTelephoneNumber** property is a **PtypString** value that contains a mail user's radio telephone number.

2.2.4.47 PidTagCarTelephoneNumber

The **PidTagCarTelephoneNumber** property is a **PtypString** value that contains a mail user's car telephone number.

2.2.4.48 PidTagOtherTelephoneNumber

The **PidTagOtherTelephoneNumber** property is a **PtypString** value that contains a mail user's alternate telephone number.

2.2.4.49 PidTagPagerTelephoneNumber

The **PidTagPagerTelephoneNumber** property is a **PtypString** value that contains a mail user's pager telephone number.

2.2.4.50 PidTagPrimaryFaxNumber

The **PidTagPrimaryFaxNumber** property is a **PtypString** value that contains the telephone number of a mail user's primary fax machine.

2.2.4.51 PidTagBusinessFaxNumber

The **PidTagBusinessFaxNumber** property is a **PtypString** value that contains the telephone number of a mail user's business fax machine.

2.2.4.52 PidTagHomeFaxNumber

The **PidTagHomeFaxNumber** property is a **PtypString** value that contains the telephone number of a mail user's home fax machine.

2.2.4.53 PidTagCompanyMainTelephoneNumber

The **PidTagCompanyMainTelephoneNumber** property is a **PtypString** value that contains the main telephone number of a mail user's company.

2.2.4.54 PidTagTelecommunicationsDeviceForDeafTelephoneNumber

The **PidTagTelecommunicationsDeviceForDeafTelephoneNumber** property is a **PtypString** value that contains a mail user's telecommunication device for the deaf (TTY/TDD) telephone number.

2.2.4.55 PidTagTelexNumber

The **PidTagTelexNumber** property is a **PtypString** value that contains a mail user's telex number.

2.2.4.56 PidTagIsdnNumber

The **PidTagIsdnNumber** property is a **PtypString** value that contains a mail user's ISDN-capable telephone number.

2.2.4.57 PidTagAssistantTelephoneNumber

The **PidTagAssistantTelephoneNumber** property is a **PtypString** value that contains the telephone number of a mail user's administrative assistant.

2.2.4.58 PidTagKeyword

The **PidTagKeyword** property is a **PtypString** value containing a keyword that identifies a mail user to his or her system administrator.

2.2.4.59 PidTagGovernmentIdNumber

The **PidTagGovernmentIdNumber** property is a **PtypString** value that contains a mail user's government identifier.

2.2.4.60 PidTagMessageHandlingSystemCommonName

The **PidTagMessageHandlingSystemCommonName** property is a **PtypString** value that contains the **common name** of a messaging user for use in a message header. This property is read-only.

2.2.4.61 PidTagLanguage

The **PidTagLanguage** property is a **PtypString** containing a value that indicates the language in which a messaging user writes messages.

2.2.4.62 PidTagLocation

The **PidTagLocation** property is a **PtypString** value that contains the location of a mail user in a format that is useful to that person's organization.

2.2.4.63 PidTagOrganizationalIdNumber

The **PidTagOrganizationalIdNumber** property is a **PtypString** value that contains an identifier for a mail user that is used within that person's organization.

2.2.4.64 PidTagUserCertificate

The **PidTagUserCertificate** property is a **PtypBinary** value that has been deprecated. This property MUST be ignored by clients.

2.2.4.65 PidTagAddressBookX509Certificate

The **PidTagAddressBookX509Certificate** property is a **PtypMultipleBinary** value that 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

The **PidTagUserX509Certificate** property is a **PtypMultipleBinary** value that 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)																															
...																															
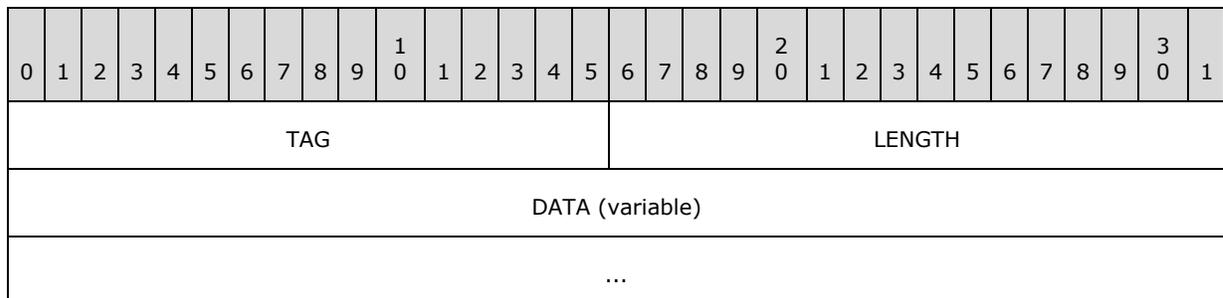
TAG (2 bytes): An unsigned value that identifies this particular security setting. If the **TAG** value found in a security setting is not among the ones listed in sections 2.2.4.66.1 through 2.2.4.66.12, the client and the server MUST ignore these settings.

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 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** 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** 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.<17>

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.<18>

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** 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.[19]

For a specified **PtypBinary** 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.

For a specified **PtypBinary** 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].[20]

For a specified **PtypBinary** 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** 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 larger than 4. The **DATA** field contains the ASN.1 DER encoded **sMIMECapabilities** type, as specified in [RFC3851]. This setting MUST be part of the **PtypBinary** 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.

2.2.4.67   PidTagAddressBookHomeMessageDatabase

The **PidTagAddressBookHomeMessageDatabase** property is a **PtypString** value that contains the distinguished name (DN) 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 is an *x500-dn* string that references an Address Book object that represents the mail user's mailbox server. The relative distinguished name (RDN) in the *object-rdn* of the *x500-dn* for the mailbox server's Address Book object is the host-name, as described in [RFC1034], of the server that contains the mail user's message database.

2.2.4.68   PidTagAddressBookNetworkAddress

The **PidTagAddressBookNetworkAddress** property is a **PtypMultipleString** value that 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

The **PidTagHobbies** property is a **PtypString** value that contains the names of a mail user's hobbies.

2.2.4.70   PidTagProfession

The **PidTagProfession** property is a **PtypString** value that contains the name of a mail user's line of business.

2.2.4.71   PidTagReferredByName

The **PidTagReferredByName** property is a **PtypString** value that contains the name of a mail user's referral.

2.2.4.72   PidTagSpouseName

The **PidTagSpouseName** property is a **PtypString** value that contains the name of a mail user's spouse/partner.

2.2.4.73   PidTagGender

The **PidTagGender** property is a **PtypInteger16** value containing 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

The **PidTagComputerNetworkName** property is a **PtypString** value that contains the name of a mail user's computer network.

2.2.4.75 PidTagCustomerId

The **PidTagCustomerId** property is a **PtypString** value that contains a mail user's customer identification number.

2.2.4.76 PidTagFtpSite

The **PidTagFtpSite** property is a **PtypString** value that contains a mail user's **File Transfer Protocol (FTP)** site address.

2.2.4.77 PidTagPersonalHomePage

The **PidTagPersonalHomePage** property is a **PtypString** value that contains the Web address (**URL**) of a mail user's personal home page.

2.2.4.78 PidTagBusinessHomePage

The **PidTagBusinessHomePage** property is a **PtypString** value that contains the Web address (URL) of a mail user's business home page.

2.2.4.79 PidTagBirthday

The **PidTagBirthday** property is a **PtypTime** value that contains the date of a mail user's birthday at 12:00 a.m. Coordinated Universal Time (UTC).

2.2.4.80 PidTagWeddingAnniversary

The **PidTagWeddingAnniversary** property is a **PtypTime** value that contains the date of a mail user's wedding anniversary at 12:00 A.M. Coordinated Universal Time (UTC).

2.2.4.81 PidTagAddressBookOrganizationalUnitRootDistinguishedName

The **PidTagAddressBookOrganizationalUnitRootDistinguishedName** property is a **PtypString** value that contains the distinguished name (DN) 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.<21>

2.2.4.82 PidTagThumbnailPhoto

The **PidTagThumbnailPhoto**<22> property is a **PtypBinary** value that, if present, MUST contain an image of ae mail user's photo in Joint Photographic Experts Group (JPEG)format.

2.2.4.83 PidTagSpokenName

The **PidTagSpokenName** property<23> is a **PtypBinary** value that, 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

The **PidTagAddressBookAuthorizedSenders** property<24> is a **PtypObject** value that indicates whether delivery restrictions exist for the specified recipient. A value other than NULL indicates that delivery restrictions exist for this recipient. The address book does not contain the lists of senders who are allowed for this recipient; it only indicates whether such restrictions exist.

2.2.4.85 PidTagAddressBookUnauthorizedSenders

The **PidTagAddressBookUnauthorizedSenders** property<25> is a **PtypObject** value that indicates whether delivery restrictions exist for the specified recipient. A value other than NULL indicates that delivery restrictions exist for this recipient. The address book does not contain the lists of senders who are prohibited for this recipient; it only indicates whether such restrictions exist.

2.2.4.86 PidTagAddressBookDistributionListMemberSubmitAccepted

The **PidTagAddressBookDistributionListMemberSubmitAccepted** property<26> is a **PtypObject** value that indicates whether delivery restrictions exist for the specified recipient. A value other than NULL indicates that delivery restrictions exist for this recipient. The address book does not contain the lists of the group of senders who are allowed for this recipient; it only indicates whether such restrictions exist.

2.2.4.87 PidTagAddressBookDistributionListMemberSubmitRejected

The **PidTagAddressBookDistributionListMemberSubmitRejected** property<27> is a **PtypObject** value that indicates whether delivery restrictions exist for this recipient. A value other than NULL indicates that delivery restrictions exist for this recipient. The address book does not contain the lists of the group of senders who are prohibited for this recipient; it only indicates whether such restrictions exist.

2.2.5 Properties That Reference Other Address Book Objects

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] section 3.1.1.3.2.2. Each of these properties is of type **PtypEmbeddedTable**, as specified in [MS-NSPI]. 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** call, and are modified through the **NspiModLinkAtt** call, as specified in [MS-NSPI].

These tables are represented in an offline address book (OAB) as a property either of type **PtypString** (for tables that consist of no more than one reference) or of type **PtypMultipleString**

(for tables that can consist of multiple references), as specified in [MS-OXOAB]. Each string value represents the distinguished name (DN) 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 of the manager.

2.2.5.1 PidTagAddressBookManager

The **PidTagAddressBookManager** property is a **PtypEmbeddedTable** value containing one row that references a mail user's manager.

2.2.5.2 PidTagAddressBookReports

The **PidTagAddressBookReports** property is a **PtypEmbeddedTable** value of a mail user that lists all of the mail user's direct reports.

2.2.5.3 PidTagAddressBookIsMemberOfDistributionList

The **PidTagAddressBookIsMemberOfDistributionList** property is a **PtypEmbeddedTable** value of an Address Book object that lists all the distribution lists for which a specified object is a member.

2.2.5.4 PidTagAddressBookOwnerBackLink

The **PidTagAddressBookOwnerBackLink** property is a **PtypEmbeddedTable** value of a mail user list that contains the distribution lists that the mail user owns.

2.2.5.5 PidTagAddressBookPublicDelegates

The **PidTagAddressBookPublicDelegates** property is a **PtypEmbeddedTable** value that contains a list of mail users who are allowed to send mail on behalf of the mailbox owner.

2.2.5.6 PidTagAddressBookHierarchicalShowInDepartments

The **PidTagAddressBookHierarchicalShowInDepartments** property is a **PtypEmbeddedTable** value of 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

The **PidTagAddressBookMember** property is a **PtypEmbeddedTable** value that 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

The **PidTagAddressBookOwner** property is a **PtypEmbeddedTable** value containing one row that references a distribution list's owner.

2.2.6.3   PidTagContainerContents

The **PidTagContainerContents** property is a **PtypEmbeddedTable** value containing 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

The **PidTagAddressBookFolderPathname** property is a **PtypString** value that has been deprecated and MUST be ignored by clients.

2.2.6.5   PidTagAddressBookHierarchicalIsHierarchicalGroup

The **PidTagAddressBookHierarchicalIsHierarchicalGroup** property is a **PtypBoolean** value that contains "TRUE" 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" 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.<28><29> 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 distinguished name (DN), 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 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

The **PidTagAddressBookRoomContainers** property is a **PtypMultipleString** value that contains a list of distinguished names (DNs) 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 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 of the *object-rdn* follows the *container-guid* format specification. When the relative distinguished name (RDN) is extracted from the DN, 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, 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

The **PidTagAddressBookHierarchicalRootDepartment** property is a **PtypEmbeddedTable** value that contains a reference to either the root Department object or the root departmental group in the department hierarchy for an organization.<30> 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

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<31> 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 distinguished name (DN) 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

The **PidTagAddressBookHierarchicalChildDepartments** property is a **PtypEmbeddedTable** value containing a Department object that references the child departments in a hierarchy of departments.

2.2.8.2 PidTagAddressBookHierarchicalParentDepartment

The **PidTagAddressBookHierarchicalParentDepartment** property is a **PtypEmbeddedTable** value containing a Department object that references all of the departments to which this specified department is a child department.

2.2.8.3 PidTagAddressBookHierarchicalDepartmentMembers

The **PidTagAddressBookHierarchicalDepartmentMembers** property is a **PtypEmbeddedTable** value containing 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

The **PidTagAddressBookRoomCapacity** property is a **PtypInteger32** value that contains the maximum occupancy of a specified room.

2.2.9.2 PidTagAddressBookRoomDescription

The **PidTagAddressBookRoomDescription** property is a **PtypString** value that contains a description of a Resource object.

2.2.10   Properties That Have Special Purposes

2.2.10.1   PidTagAnr

The **PidTagAnr** property is a **PtypString** value containing 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 NSPI call **NspiGetMatches** (as specified in [MS-NSPI]).<32> Messaging clients pass this property as a target string to the **NspiGetMatches** 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]. 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.<33>

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

The **PidTagAddressBookManageDistributionList** object property is a **property tag** for use in display templates for distribution lists. When **PidTagAddressBookManageDistributionList** 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**. Offline address books (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

The address book contains one or more address book containers. The collection of address book containers is arranged in an address book hierarchy table. An address book container represents an address list, which is a collection of Address Book objects that are rendered in a table and browsed by a messaging user. Among other things, 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 address list. ANR means that the user types part of a name, and the address book identifies potential matches for that name.

Messaging users are able to display information about an Address Book object from the table, an object obtained through ANR, or a recipient on a message. The information displayed is obtained from a display template that is supplied in the address book, and the type of information depends on the type of Address Book object being displayed.

Among the many types of objects, an address book typically includes the following object types:

- 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 Address Book objects 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.

- Address book containers, each of which represents an address list that contains Address Book objects that can be viewed as a table.

- 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].

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

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

3.1.3.2 Initialization Through an Offline Address Book

Messaging clients obtain address lists in an OAB by using the Offline Address Book (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 Offline Address Book (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

When using an NSPI server, messaging clients obtain the hierarchy by using the **NspiGetSpecialTable** method, as specified in [MS-NSPI]. The table returns several rows of containers, and for each row, it returns the properties **PidTagDisplayName** (section 2.2.3.1), **PidTagEntryId** (section 2.2.3.2), **PidTagContainerFlags** (section 2.2.2.1), **PidTagDepth** (section 2.2.2.2), **PidTagAddressBookContainerId** (section 2.2.2.3), **PidTagAddressBookIsMaster** (section 2.2.2.4), and **PidTagAddressBookParentEntryId** (section 2.2.2.5), which are needed to determine the hierarchy of containers.

The hierarchy table in an OAB is obtained by using the Offline Address Book (OAB) Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB].

3.1.4.2 Browsing an Address Book

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 Offline Address Book (OAB) Format and Schema Protocol, as specified in [MS-OXOAB].

When browsing an address list from an NSPI server, messaging clients use the **NspiQueryRows** call to obtain a set of rows to display to the messaging user. Messaging clients choose the properties they want to render; typically they would request **PidTagEntryId** (section 2.2.3.2), **PidTagDisplayName** (section 2.2.3.1), **PidTagSmtpAddress** (section 2.2.3.21), and **PidTagTitle** (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 then calling the **NspiUpdateStat** method. Messaging clients usually follow up by calling the **NspiQueryRows** method 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** call, as specified in [MS-NSPI]. The result updates positioning information in the **STAT** structure, as specified in [MS-NSPI], and returns a screen full of Address Book objects.

3.1.4.3   Obtaining Properties on an Address Book Object

To obtain properties for an Address Book object from an OAB, the messaging client needs to have either a distinguished name (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 Offline Address Book (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 **NspiQueryRows**, **NspiSeekEntries**, or **NspiGetMatches** — include the DN(1) or Minimal Entry ID in their **PidTagEntryId** calls, or as a return value of the calsl. Clients call **NspiDNToMId** 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** or **NspiResolveNamesW** call, as specified in [MS-NSPI], and extract the DN from the **PidTagEntryId** (section 2.2.3.2) property that is returned, as specified in the Permanent Entry ID structure, as specified in [MS-NSPI].

After the Minimal Entry ID is known, clients use it as the **CurrentRec** member of the **STAT** structure passed to **NspiGetProps** method, as specified in [MS-NSPI]. It is possible for clients to specify any number of properties in the *pPropTags* parameter to **NspiGetProps** 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 PidTagBusinessTelephoneNumber** (section 2.2.4.38) and **PidTagSmtpAddress** (section 2.2.3.21) in the *pPropTags* parameter.

3.1.4.4   Performing Ambiguous Name Resolution

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 Offline Address Book (OAB) Format and Schema Protocol, as specified in [MS-OXOAB].

When performing ANR by using an NSPI server, the client calls the **NspiResolveNames** method or the **NspiResolveNamesW** method and passes the target string in the parameter *paStr* or *paWStr*, as specified in [MS-NSPI]. 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], by using a *Filter* parameter with the **PidTagAnr** property and using the value for **PidTagAnr** as the target string for ANR.<34> The set of matches, along with the requested properties for the matches, will be returned. Messaging clients usually 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

A messaging client uses an offline address book (OAB), an NSPI server, or both, to access an address book. A client obtains one or more address lists from an OAB by using the Offline Address Book Web and Public Folder Retrieval protocols, as specified in [MS-OXWOAB] and in [MS-OXPFOAB]. Usually, the client downloads the contents of address lists from the OAB into one or more files onthe local computer. Messaging clients access information about the Address Book objects that are contained in an address list from an OAB by using the Offline Address Book (OAB) Format and Schema Protocol, as specified in [MS-OXOAB]. Alternatively, the client accesses information about objects in an address book by using an NSPI server, using the Name Service Provider Interface (NSPI) Protocol, as specified in [MS-NSPI]. Because both data sources represent the same address book, information about an Address Book object that is contained on both an NSPI server and in an OAB SHOULD<35> be identical, with the OAB containing a subset of the information that is available on the NSPI server.<36> Because the **PidTagEmailAddress** property (section 2.2.3.14), which matches the DN(1) for an Address Book object, is used to identify that object, the value for **PidTagEmailAddress** MUST match for an object that is contained in both data sources.

The role of the server that contains an OAB is to supply all of the necessary information about the Address Book objects by using the Offline Address Book (OAB) Format and Schema Protocol, as specified in [MS-OXOAB], and deliver that information to the client by using the Offline Address Book (OAB) Public Folder Retrieval Protocol, as specified in [MS-OXPFOAB]. The server provides periodic updates to the OAB, as necessary. This updating mechanism is specified in [MS-OXOAB]. After the information is delivered to the client, the server plays no further role.

The role of the NSPI server is to service the requests of the client, by using the Name Service Provider Interface (NSPI) Protocol, as specified in [MS-NSPI].

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

None.

3.2.6 Timer Events

None.

3.2.7 Other Local Events

None.

4 Protocol Examples

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 distinguished name (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.
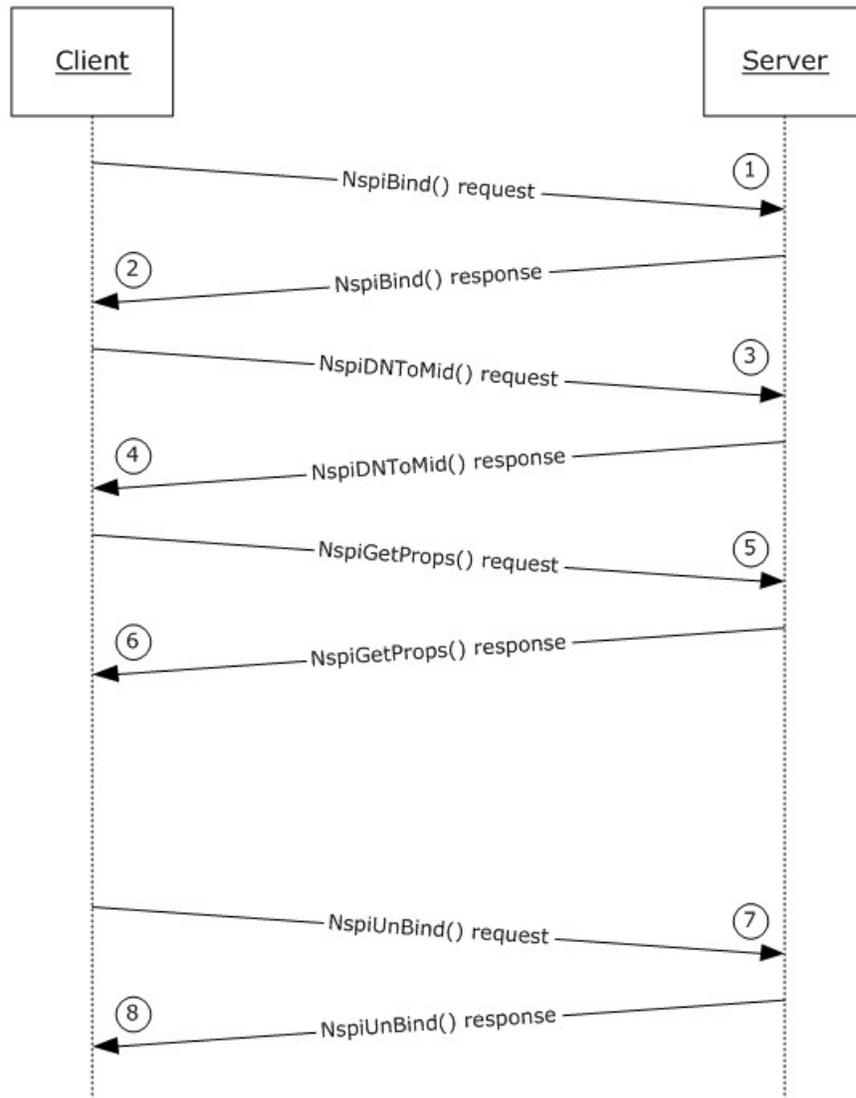
Figure 1: Call sequence between client and server

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

- 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

0x00000000DWORD

pStat

hIndex0x00000000unsigned long

ContainerID0x00000000unsigned long

CurrentRec0x00000000unsigned long

Delta0x00000000long

NumPos0x00000000unsigned long

TotalRecs0x00000000unsigned long

CodePage0x000004e4unsigned long

TemplateLocale0x00000409unsigned long

SortLocale0x00000409unsigned long

pServerGuid

< a pointer to an array of 16 unsigned chars to be returned by the server>

- The server responds to the **NspiBind** 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>

- The client requests the Minimal Entry ID for the distinguished name (DN)(1) "/o=First Organization/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/CN=Recipients/CN=user1" by calling the **NspiDNToMId** method with parameters that typically are as follows:

_pNames

Count0x00000001DWORD

Stringschar **

[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>

▪ The server responds with return code "Success" with a value of "0x00001927" for a Minimal Entry ID that typically is as follows:

Count0x00000001DWORD

Stringschar **

[0x0]char *

"/o=First Organization/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/cn=Recipients/cn=user1"

ppMIds

cValues   0x00000001   DWORD

aulPropTag   DWORD[]

[0x0]

0x00001927   DWORD

▪ 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 parameters that typically are as follows:

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

hRpc< a valid RPC handle>void *

dwFlags0x00000000DWORD

pStat

hIndex0x00000000unsigned long

ContainerID0x00000000unsigned long

CurrentRec0x00001927unsigned long

Delta0x00000000long

NumPos0x00000000unsigned long

TotalRecs0x00000000unsigned long

CodePage0x000004b0unsigned long

TemplateLocale0x00000409unsigned long

SortLocale0x00000409unsigned long

pPropTags_SPropTagArray_r *

{

cValues0x00000002 DWORD

aulPropTag=<a pointer to an array of proptags>

}

aulPropTagunsigned long []

[0x0]PidTagDisplayName unsigned long

[0x1]PidTagGivenName unsigned long

ppRow_SRow_r * *

< memory location for server return values>

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

dwFlags0x00000000DWORD

pStat

{

hIndex0x00000000unsigned long

ContainerID0x00000000unsigned long

CurrentRec0x00001927unsigned long

Delta0x00000000long

NumPos0x00000000unsigned long

TotalRecs0x00000000unsigned long

CodePage0x000004b0unsigned long

TemplateLocale0x00000409unsigned long

SortLocale0x00000409unsigned long

}

pPropTags_SPropTagArray_r *

{

cValues0x00000002 DWORD

aulPropTag=<a pointer to an array of proptags>

}

aulPropTagunsigned long []

[0x0]PidTagDisplayName unsigned long

[0x1]PidTagGivenName unsigned long

ppRows_SRowSet_r * *

{

cRows0x00000001DWORD

aRow=<a pointer to an array of rows>

}

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

aRow[0x0]_SRow_r *

{

cValues0x00000002DWORD

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]

ulPropTagPidTagDisplayName 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.

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

contextHandleNSPI_HANDLE *

<a token which was sent by the server in the NspiBind call >

dwFlags

0x00000000unsigned long

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

5   Security

5.1   Security Considerations for Implementers

There are no special security considerations specific to the Address Book Object Protocol. Note, however, that general security considerations pertaining to the underlying NSPI RPC-based transport, as specified in [MS-NSPI], 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® Office Outlook® 2003

- Microsoft® Office Outlook® 2007

- Microsoft® Outlook® 2010

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

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

<1> Section 2.2.1.1: The OAB that is implemented in Exchange 2003, Exchange 2007, and Exchange 2010 contains all the properties listed in the appendixes of [MS-OXOAB]. This can include some properties that are not present on some implementations of an NSPI server.

<2> Section 2.2.1.1: The OAB version 4 implementation in Exchange 2003, Exchange 2007, and Exchange 2010 includes a **PidTagOfflineAddressBookTruncatedProperties** property that is not included in objects on a NSPI server. This property is specified in [MS-OXOAB].

<3> Section 2.2.1.1: The values for string and binary properties might be truncated in an OAB according to the limitations specified in [MS-OXOAB].

<4> Section 2.2.1.1: The properties in an OAB might have a different value if the value on a 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.

<5> Section 2.2.3: The Exchange 2003, Exchange 2007, and Exchange 2010 SP1 servers include the additional properties specified in section 2.2.6. Specific properties can be restricted from access by clients, such as a distribution lists in which the members are hidden.

<6> Section 2.2.3.9: The **PidTagAddressBookPhoneticDisplayName** property is not used in Exchange 2003 servers or Office Outlook 2003 clients.

<7> Section 2.2.3.26: The **PidTagAddressBookSenderHintTranslations** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<8> Section 2.2.3.27: The **PidTagAddressBookDeliveryContentLength** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<9> Section 2.2.3.28: The **PidTagAddressBookModerationEnabled** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<10> Section 2.2.3.29: The **PidTagAddressBookDistributionListMemberCount** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<11> Section 2.2.3.30: The **PidTagAddressBookDistributionListExternalMemberCount** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<12> Section 2.2.3.35: The **PidTagAddressBookDisplayTypeExtended** property is not used in Exchange 2003 servers or Office Outlook 2003 clients.

<13> Section 2.2.4.14: The **PidTagAddressBookPhoneticGivenName** property is not used in Exchange 2003 servers or Office Outlook 2003 clients.

<14> Section 2.2.4.15: The **PidTagAddressBookPhoneticSurname** property is not used in Exchange 2003 servers or Office Outlook 2003 clients.

<15> Section 2.2.4.16: The **PidTagAddressBookPhoneticCompanyName** property is not used by Exchange 2003 servers or Office Outlook 2003 clients.

<16> Section 2.2.4.17: The **PidTagAddressBookPhoneticDepartmentName** property is not used by Exchange 2003 servers or Office Outlook 2003 clients.

<17> Section 2.2.4.66.4: The **PidTagAddressBookPhoneticDisplayName** property is not used in Exchange 2003 or Office Outlook 2003 servers.

<18> Section 2.2.4.66.5: The **PidTagAddressBookPhoneticDisplayName** property is not used in Exchange 2003 or Office Outlook 2003 servers.

<19> Section 2.2.4.66.7: The **PidTagAddressBookPhoneticDisplayName** property is not used in Exchange 2003 or Office Outlook 2003 servers.

<20> Section 2.2.4.66.9: The **PidTagAddressBookPhoneticDisplayName** property is not used in Exchange 2003 or Office Outlook 2003 servers.

<21> Section 2.2.4.81: The **PidTagAddressBookOrganizationalUnitRootDistinguishedName** property is not used in Office Outlook 2003 clients.

<22> Section 2.2.4.82: The **PidTagThumbnailPhoto** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<23> Section 2.2.4.83: The **PidTagSpokenName** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<24> Section 2.2.4.84: The **PidTagAddressBookAuthorizedSenders** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<25> Section 2.2.4.85: The **PidTagAddressBookUnauthorizedSenders** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<26> Section 2.2.4.86: The **PidTagAddressBookDistributionListMemberSubmitAccepted** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<27> Section 2.2.4.87: The **PidTagAddressBookDistributionListMemberSubmitRejected** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<28> Section 2.2.7: The **PidTagAddressBookOrganizationalUnitRootDistinguishedName** property is not used in Office Outlook 2003 clients.

<29> Section 2.2.7: The **PidTagAddressBookSenderHintTranslations** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<30> 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 is not present, and there is no hierarchical representation of the departments.

<31> 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 and Outlook 2010 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.

<32> Section 2.2.10.1: Office Outlook 2003, Office Outlook 2007, and Outlook 2010 specify an ambiguous name resolution search by setting up a *Filter* parameter to the **NspiGetMatches** method, as specified in [MS-NSPI]. The *Filter* parameter is a **Restriction_r** structure, whose members have the values that are listed in the following table.

Member	Value
Filter.rt	0x00000004 (same as **RES_PROPERTY** as specified in [MS-OXCDATA]).
Filter.res.resProperty.relop	0x00000004 (same as **RELOP_EQ** as specified in [MS-OXCDATA]).
Filter.res.resProperty.ulPropTag	**PidTagAnr** (**PtypString8** or **PtypString**).
Filter.res.resProperty.lpProp.ulPropTag	Same as Filter.res.resProperty.ulPropTag.
Filter.res.resProperty.lpProp.Value.lpszA	Target string in the code page of pStat.CodePage, if **PtypString8** is specified.
Filter.res.resProperty.lpProp.Value.lpszW	Target string in Unicode, if **PtypString** is specified.

<33> Section 2.2.10.1: Office Outlook 2003, Office Outlook 2007, and Outlook 2010 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, and Exchange 2010 SP1 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 or the *paWStr* parameter to the **NspiResolveNamesW** method.

<34> Section 3.1.4.4: The **PidTagAddressBookSenderHintTranslations** property is not used by Exchange 2003, Exchange 2007, Office Outlook 2003, or Office Outlook 2007.

<35> Section 3.2.1: The properties in an offline address book (OAB) might have a different value 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.

<36> Section 3.2.1: The offline address book (OAB) that is implemented in Exchange 2003, Exchange 2007, and Exchange 2010 contains all of the properties listed in the appendixes in [MS-OXOAB]. This can include some properties that are not present on some implementations of an NSPI server.

7 Change Tracking

This section identifies changes that were made to the [MS-OXOABK] protocol document between the November 2010 and March 2011 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

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

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

- A document revision that incorporates changes to interoperability requirements or functionality.

- An extensive rewrite, addition, or deletion of major portions of content.

- The removal of a document from the documentation set.

- Changes made for template compliance.

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

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

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

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

- New content added.

- Content updated.

- Content removed.

- New product behavior note added.

- Product behavior note updated.

- Product behavior note removed.

- New protocol syntax added.

- Protocol syntax updated.

- Protocol syntax removed.

- New content added due to protocol revision.

- Content updated due to protocol revision.

- Content removed due to protocol revision.

- New protocol syntax added due to protocol revision.

- Protocol syntax updated due to protocol revision.

- Protocol syntax removed due to protocol revision.

- New content added for template compliance.

- Content updated for template compliance.

- Content removed for template compliance.

- Obsolete document removed.

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

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

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.

- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact protocol@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
1 Introduction	Added information about which sections of the specification are normative and can contain RFC 2119 language.	Y	New content added for template compliance.

8 Index

Release: Tuesday, March 15, 2011