

[MS-ASNOTE]: ActiveSync Notes Class Protocol Specification

Intellectual Property Rights Notice for Open Specifications Documentation

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

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

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

Revision Summary

Date	Revision History	Revision Class	Comments
04/10/2009	0.1.0	Major	Initial Availability.
07/15/2009	1.0.0	Major	Revised and edited for technical content.
11/04/2009	2.0.0	Major	Updated and revised the technical content.
02/10/2010	3.0.0	Major	Updated and revised the technical content.
05/05/2010	4.0.0	Major	Updated and revised the technical content.
08/04/2010	4.1	Minor	Clarified the meaning of the technical content.
11/03/2010	4.2	Minor	Clarified the meaning of the technical content.

Contents

1 Introduction	5
1.1 Glossary	5
1.2 References	5
1.2.1 Normative References	5
1.2.2 Informative References	6
1.3 Overview	6
1.4 Relationship to Other Protocols	6
1.5 Prerequisites/Preconditions	6
1.6 Applicability Statement	6
1.7 Versioning and Capability Negotiation	6
1.8 Vendor-Extensible Fields	6
1.9 Standards Assignments	6
2 Messages	7
2.1 Transport	7
2.2 Message Syntax	7
2.2.1 Namespaces	8
2.2.2 Elements	8
2.2.2.1 airsynbase:Body	8
2.2.2.2 Subject	9
2.2.2.3 MessageClass	9
2.2.2.4 LastModifiedDate	9
2.2.2.5 Categories	9
2.2.2.5.1 Category	9
3 Protocol Details	10
3.1 Client Details	10
3.1.1 Abstract Data Model	10
3.1.2 Timers	10
3.1.3 Initialization	10
3.1.4 Higher-Layer Triggered Events	10
3.1.4.1 Synchronizing Notes Data with a Server	10
3.1.4.2 Searching a Server for Notes	10
3.1.4.3 Requesting Details for One or More Notes	10
3.1.5 Message Processing Events and Sequencing Rules	10
3.1.5.1 ItemOperations Command Request	10
3.1.5.1.1 airsynbase:Body Element	11
3.1.5.1.2 MessageClass Element	11
3.1.5.2 Search Command Request	11
3.1.5.3 Sync Command Request	11
3.1.5.3.1 LastModifiedDate Element	11
3.1.6 Timer Events	11
3.1.7 Other Local Events	11
3.2 Server Details	12
3.2.1 Abstract Data Model	12
3.2.2 Timers	12
3.2.3 Initialization	12
3.2.4 Higher-Layer Triggered Events	12
3.2.4.1 Synchronizing Notes Data with a Server	12
3.2.4.2 Searching a Server for Notes	12

3.2.4.3	Requesting Details for One or More Notes	12
3.2.5	Message Processing Events and Sequencing Rules	12
3.2.5.1	ItemOperations Command Response	12
3.2.5.2	Search Command Response	13
3.2.5.3	Sync Command Response	13
3.2.5.3.1	LastModifiedDate Element	13
4	Protocol Examples	14
5	Security	16
5.1	Security Considerations for Implementers	16
5.2	Index of Security Parameters	16
6	Appendix A: Product Behavior	17
7	Change Tracking	18
8	Index	20

1 Introduction

This document specifies the Notes **class** protocol, which facilitates a mobile device synchronizing user notes with a server that supports the ActiveSync Protocol.

1.1 Glossary

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

class
collection
HTML
Inter-Personal Mail (IPM)
plain text
property
Rich Text Format (RTF)
synchronization
Wireless Application Protocol (WAP) Binary XML (WBXML)
XML
XML namespace
XML schema
XML schema definition (XSD)

The following terms are specific to this document:

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

1.2 References

1.2.1 Normative References

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

[MS-ASAIRS] Microsoft Corporation, "[ActiveSync AirSyncBase Namespace Protocol Specification](#)", December 2008.

[MS-ASCMD] Microsoft Corporation, "[ActiveSync Command Reference Protocol Specification](#)", December 2008.

[MS-ASDTYPE] Microsoft Corporation, "[ActiveSync Data Types](#)", December 2008.

[MS-ASWBXML] Microsoft Corporation, "[ActiveSync WAP Binary XML \(WBXML\) Protocol Specification](#)", December 2008.

[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>

[XML] World Wide Web Consortium, "Extensible Markup Language (XML) 1.0 (Fourth Edition)", W3C Recommendation, 16 August 2006, edited in place 29 September 2006, <http://www.w3.org/TR/2006/REC-xml-20060816/>

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

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

1.2.2 Informative References

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

1.3 Overview

The Notes class protocol specifies the **XML** representation of notes used for client and server communication, as specified in [\[MS-ASCMD\]](#).

1.4 Relationship to Other Protocols

The Notes class protocol specifies the XML representation of notes that are used by commands specified in [\[MS-ASCMD\]](#). The protocol governing the transmission of these commands between the client and the server is specified in [\[MS-ASCMD\]](#).

All simple data types in this document conform to the data type definitions specified in [\[MS-ASDTYPE\]](#).

1.5 Prerequisites/Preconditions

None.

1.6 Applicability Statement

This protocol specifies a set of elements for use in communicating note data using the commands specified in [\[MS-ASCMD\]](#). This set of elements is applicable when communicating note information between a mobile device and a server. Except where indicated, these elements are not applicable when sending other types of information supported by the ActiveSync protocol.

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

The Notes class consists of a series of XML elements that are embedded inside of a command or a **collection** sent in accordance with [\[MS-ASCMD\]](#). The XML block containing the class elements is transmitted in either the request body of a request, or the response body of a response. The parent element of the Notes class elements depends upon the ActiveSync protocol command used to retrieve the class data. Commands and parent elements for the Notes class **XML schema** are specified in section [3.1.5](#).

The elements of the Notes class are defined in two namespaces: Note, whose elements are specified in this document, and AirSyncBase, whose elements are specified in [\[MS-ASAIRS\]](#).

2.2 Message Syntax

The markup MUST be well-formed XML, as specified in [\[XML\]](#).

The XML markup that constitutes the request body or the response body is transmitted between the client and the server using **Wireless Application Protocol (WAP) Binary XML (WBXML)**. For more details, see [\[MS-ASWBXML\]](#).

The **XML schema definition (XSD)** for the Notes class in ActiveSync is defined in accordance with the rules specified in [\[XMLSCHEMA1\]](#) as follows. The following represents the full set of data that can be returned by the **Sync** command.

```
<?xml version="1.0" ?><xs:schema xmlns:tns="Note:" attributeFormDefault="unqualified"
elementFormDefault="qualified"
targetNamespace="Note:" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:A="AirSyncBase:">
  <xs:import namespace="AirSyncBase:" location="AirSyncBase.xsd"/>
  <xs:element name="Subject" type="xs:string" />
  <xs:element name="MessageClass" type="xs:string" />
  <xs:element name="LastModifiedDate" type="xs:dateTime" />
  <xs:element name="Categories">
    <xs:complexType>
      <xs:sequence minOccurs="0">
        <xs:element maxOccurs="300" name="Category" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

The portion of the AirSyncBase namespace used by the Notes class is defined as follows. For the complete AirSyncBase XSD, see [\[MS-ASAIRS\]](#) section 2.2.

```
<?xml version="1.0" encoding="utf-8"?><xs:schema xmlns:airsyncbase="AirSyncBase:"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns="AirSyncBase:"
targetNamespace="AirSyncBase:" elementFormDefault="qualified"
attributeFormDefault="unqualified"> <xs:element name="Body"/>
</xs:schema>
```

2.2.1 Namespaces

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

Prefix	Namespace URI	Reference
airsynbase	AirSyncBase	[MS-ASAIRS]
None	Notes	
airsync	AirSync	[MS-ASCMD] section 2.2.2.19
itemoperations	ItemOperations	[MS-ASCMD] section 2.2.2.8
search	Search	[MS-ASCMD] section 2.2.2.14
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1]

2.2.2 Elements

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

Element	Description
airsynbase:Body	The text of the note.
Subject	The subject of the note.
MessageClass	The form of the message.
LastModifiedDate	The day and time that this note was last changed by the user.
Categories	A collection of categories for this note.
Category	One of the user-assigned labels applied to this note.

2.2.2.1 airsynbase:Body

The <airsynbase:Body> element is an optional element that specifies the text of the note. It is defined as an element in the AirSyncBase namespace.

The <airsynbase:Type> element ([\[MS-ASAIRS\]](#) section 2.2.2.2.1) of the <airsynbase:Body> element MUST be set to one of the following values.

Value	Description
1	Plain text
2	HTML
3	Rich Text Format (RTF)

When the <airsynbase:Body> element is used in a **Sync** command request or response, the <airsynbase:Data> element ([\[MS-ASAIRS\]](#) section 2.2.2.4.4) of the <airsynbase:Body> element is required.

For more details about <airsynbase:Body>, see [\[MS-ASAIRS\]](#) section 2.2.2.4.

2.2.2.2 Subject

The <Subject> element is an optional element that specifies the subject of the note.

2.2.2.3 MessageClass

The <MessageClass> element is a required element that specifies the **Inter-Personal Mail (IPM)** type of the note.

The value of the <MessageClass> element MUST be either "IPM.StickyNote" or "IPM.StickyNote.*", where "*" represents an arbitrary string chosen by the client or server.

2.2.2.4 LastModifiedDate

The <LastModifiedDate> element is an element that specifies when the note was last modified.

The role of this element in client requests and server responses is specified in sections [3.1.5.3.1](#) and [3.2.5.3.1](#).

The value of the <LastModifiedDate> element is a **dateTime** type, as specified in [\[MS-ASDTYPE\]](#) section 2.3.

2.2.2.5 Categories

The <Categories> element is an optional **container** ([\[MS-ASDTYPE\]](#) section 2.2) element that specifies a collection of categories for this note.

The <Categories> element can only have the following child element:

- <Category> (section [2.2.2.5.1](#)): Zero or more instances of this element are allowed.

2.2.2.5.1 Category

The <Category> element is an optional child element of the <Categories> element (section [2.2.2.5](#)) that specifies that a user-selected label has been applied to this note.

3 Protocol Details

3.1 Client Details

3.1.1 Abstract Data Model

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

Notes class: A structured XML text block that adheres to the XML schema definition specified in section 2.2. It is included by the server as part of a full XML response to the client commands specified in section 3.1.5. Notes class data is included in command requests sent to the server when notes need to be retrieved or synchronized. For more details about processing command requests, see section 3.1.5.

Command request: A WBXML formatted message that adheres to the command schemas specified in [MS-ASCMD].

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Higher-Layer Triggered Events

3.1.4.1 Synchronizing Notes Data with a Server

A client initiates **synchronization** of Notes class data with the server by sending a **Sync** command request.

3.1.4.2 Searching a Server for Notes

A client searches for Notes class data by sending a **Search** command request to the server.

3.1.4.3 Requesting Details for One or More Notes

Notes class data for one or more individual notes is requested by the client using the **ItemOperations** command, which is a wrapper for the <itemoperations:Fetch> element. An **ItemOperations** command can contain multiple <itemoperations:Fetch> elements.

3.1.5 Message Processing Events and Sequencing Rules

3.1.5.1 ItemOperations Command Request

A client uses the **ItemOperations** command to retrieve specific Notes items from the server.

Notes class elements cannot be included in an **ItemOperations** command request.

For more details about the **ItemOperations** command, see [\[MS-ASCMD\]](#) section 2.2.2.8.

3.1.5.1.1 **airsynbase:Body Element**

If a client cannot display the data type specified by the <airsynbase:Type> element ([\[MS-ASAIRS\]](#) section 2.2.2.4.1) of the <airsynbase:Body> element (section [2.2.2.1](#)), then the client can ignore the <airsynbase:Body> element.

3.1.5.1.2 **MessageClass Element**

Clients MUST NOT change the value of the <MessageClass> element on an existing note to any value other than the required values specified in section [2.2.2.3](#). If the client submits a value other than the values specified in section [2.2.2.3](#) for this element in a <Sync> command request, then the server MUST respond with a status error of 6.

3.1.5.2 **Search Command Request**

A client uses the **Search** command to retrieve Notes class items that match the criteria specified by the client.

The elements for the Notes class MUST NOT be included in a **Search** command request or the server will return a protocol error.

For more details about the **Search** command, see [\[MS-ASCMD\]](#) section 2.2.2.14.

3.1.5.3 **Sync Command Request**

A client uses the **Sync** command to synchronize its Notes class items for a specified user with the notes currently stored by the server.

Any of the elements for the Notes class can be included in a **Sync** command request as children of the <airsync:ApplicationData> element in either an <airsync:Add> element request ([\[MS-ASCMD\]](#) section 2.2.2.19.1.2.1.1.9.3.3) or an <airsync:Change> element request ([\[MS-ASCMD\]](#) section 2.2.2.19.1.2.1.1.9.1.2).

The <airsync:Supported> element MUST NOT be included in a **Sync** command request for the Notes class. If it is included, the server will return a status error 4 (Protocol Error).

When an existing note is updated with an <airsync:Change> element in a **Sync** command, the command will contain all required elements of the note.

For more details about the **Sync** command, see [\[MS-ASCMD\]](#) section 2.2.2.19.

3.1.5.3.1 **LastModifiedDate Element**

The <LastModifiedDate> element can be excluded from a client request. If it is included in a client request, then the server will ignore it.

3.1.6 **Timer Events**

None.

3.1.7 **Other Local Events**

None.

3.2 Server Details

3.2.1 Abstract Data Model

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

Notes class: A structured XML text block that adheres to the XML schema defined in section [2.2](#). It is returned by the server as part of a full XML response to the client commands specified in section [3.1.5](#).

The server can return zero or more Notes class blocks in its response, depending on how many notes match the criteria specified by the client command request.

The server returns a Notes class XML block for every note that matches the criteria specified by the client command request.

Command response: A WBXML formatted message that adheres to the command schemas specified in [\[MS-ASCMD\]](#).

3.2.2 Timers

None

3.2.3 Initialization

None.

3.2.4 Higher-Layer Triggered Events

3.2.4.1 Synchronizing Notes Data with a Server

A client initiates synchronization of Notes class data with the server by sending a **Sync** command request.

3.2.4.2 Searching a Server for Notes

A client searches for Notes class data by sending a **Search** command request to the server.

3.2.4.3 Requesting Details for One or More Notes

Notes class data for one or more individual notes is requested by the client using the **ItemOperations** command, which is a wrapper for the <itemoperations:Fetch> element. An **ItemOperations** command can contain multiple <itemoperations:Fetch> elements.

3.2.5 Message Processing Events and Sequencing Rules

3.2.5.1 ItemOperations Command Response

A client uses the **ItemOperations** command to retrieve specific Notes items from the server.

Any of the elements for the Notes class can be included in an **ItemOperations** command response. If an <itemoperations:Schema> element was included in the command request, then the elements returned MUST be restricted to the elements included in the command request's <ItemOperations> element.

Notes class elements are returned as children of the <itemoperations:Properties> element ([\[MS-ASCMD\]](#) section 2.2.2.8.3.1.2.3.6).

For more details about the **ItemOperations** command, see [\[MS-ASCMD\]](#) section 2.2.2.8.

3.2.5.2 Search Command Response

A client uses the **Search** command to retrieve Notes class items that match the criteria specified by the client.

Any of the elements for the Notes class can be included in a **Search** command response.

Notes class elements are returned as children of the <search:Properties> element ([\[MS-ASCMD\]](#) section 2.2.2.14.2.1.2.1.2.4).

For more details about the **Search** command, see [\[MS-ASCMD\]](#) section 2.2.2.14.

3.2.5.3 Sync Command Response

A client uses the **Sync** command to synchronize its Notes class items for a specified user with the notes currently stored by the server.

Any of the elements for the Notes class can be included in a **Sync** command response as children of the <airsync:ApplicationData> element in either an <airsync:Add> element response ([\[MS-ASCMD\]](#) section 2.2.2.19.2.1.3.1.4.4.2) or an <airsync:Change> element response ([\[MS-ASCMD\]](#) section 2.2.2.19.2.1.3.1.4.3.3).

When an existing note is updated with an <airsync:Change> element in a **Sync** request, the command will contain all required elements of the note. If any element that was previously set is missing, then the server will delete that **property** from the note. The only exception is the <airsyncbase:Body> element (section [2.2.2.1](#)), whose absence in an <airsync:Change> element is not to be interpreted as an implicit delete.

The **Sync** command is specified in [\[MS-ASCMD\]](#) section 2.2.2.19.

3.2.5.3.1 LastModifiedDate Element

The <LastModifiedDate> element is not required in the client request, but is required in a server response.

If a client request includes a <LastModifiedDate> element, then the server ignores it and returns the actual time that the note was last modified.

4 Protocol Examples

The following example shows a Sync request and response where the client is creating one note, updating a second, and deleting a third.

XML request:

```
<?xml version="1.0" encoding="utf-8"?>
<Sync
  xmlns:airsynbase="AirSyncBase:"
  xmlns:notes="Notes:" xmlns="AirSync:">
  <Collections>
    <Collection>
      <SyncKey>398434774</SyncKey>
      <CollectionId>8</CollectionId>
      <DeletesAsMoves>1</DeletesAsMoves>
      <GetChanges>1</GetChanges>
      <WindowSize>512</WindowSize>
      <Options>
        <airsynbase:BodyPreference>
          <airsynbase:Type>2</airsynbase:Type>
          <airsynbase:TruncationSize>5120</airsynbase:TruncationSize>
          <airsynbase:AllOrNone>1</airsynbase:AllOrNone>
        </airsynbase:BodyPreference>
      </Options>
    </Collection>
  </Collections>
  <Commands>
    <Add>
      <ClientId>c212ac10-0465-4983-a898-076e152552ef</ClientId>
      <ApplicationData>
        <airsynbase:Body>
          <airsynbase:Type>2</airsynbase:Type>
          <airsynbase:Data>A new note I just created.</airsynbase:Data>
        </airsynbase:Body>
        <notes:Categories>
          <notes:Category>Business</notes:Category>
        </notes:Categories>
        <notes:Subject>New note</notes:Subject>
        <notes:MessageClass>IPM.StickyNote</notes:MessageClass>
      </ApplicationData>
    </Add>
    <Delete>
      <ServerId>8:1</ServerId>
    </Delete>
    <Change>
      <ServerId>bb18e2a7-3e65-41a1-b0b2-9815539f98ad</ServerId>
      <ApplicationData>
        <airsynbase:Body>
          <airsynbase:Type>2</airsynbase:Type>
          <airsynbase:Data>&lt;strong>This is my second
note.&lt;/strong></airsynbase:Data>
        </airsynbase:Body>
        <notes:Categories>
          <notes:Category>Business</notes:Category>
        </notes:Categories>
        <notes:Subject>Second Note</notes:Subject>
        <notes:MessageClass>IPM.StickyNote</notes:MessageClass>
      </ApplicationData>
    </Change>
  </Commands>
</Sync>
```

```
    </Commands>
  </Collection>
</Collections>
</Sync>
```

XML response:

```
<?xml version="1.0" encoding="utf-8"?>
<Sync xmlns="AirSync:">
  <Collections>
    <Collection>
      <SyncKey>1960353427</SyncKey>
      <CollectionId>8</CollectionId>
      <Status>1</Status>
      <Responses>
        <Add>
          <ClientId>c212ac10-0465-4983-a898-076e152552ef</ClientId>
          <ServerId>8:3</ServerId>
          <Status>1</Status>
        </Add>
        <Change>
          <ServerId>bb18e2a7-3e65-41a1-b0b2-9815539f98ad</ServerId>
          <Status>8</Status>
        </Change>
      </Responses>
    </Collection>
  </Collections>
</Sync>
```

5 Security

5.1 Security Considerations for Implementers

None.

5.2 Index of Security Parameters

None.

6 Appendix A: Product Behavior

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

- Microsoft® Exchange Server 2010

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

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

7 Change Tracking

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change Type
3.1.5.1.2 MessageClass Element	57758 Revised guidance regarding changing the value of the "MessageClass" element.	N	Content updated.

8 Index

A

Abstract data model
[client](#) 10
[server](#) 12

C

[Change tracking](#) 18
Client
[abstract data model](#) 10

D

Data model – abstract
[client](#) 10
[server](#) 12

E

[Examples - overview](#) 14

G

[Glossary](#) 5

I

[Introduction](#) 5

M

Messages
[overview](#) 7
[syntax](#) 7
[transport](#) 7

N

[Normative references](#) 5

O

[Overview \(synopsis\)](#) 6

P

[Preconditions](#) 6
[Prerequisites](#) 6
[Product behavior](#) 17

R

References
[normative](#) 5
[Relationship to other protocols](#) 6

S

Security
[overview](#) 16
Server
[abstract data model](#) 12
Syntax
[messages - overview](#) 7

T

[Tracking changes](#) 18
[Transport](#) 7