# [MS-ASCON]:
# ActiveSync Conversations Protocol Specification

---

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

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

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

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

---

## Revision Summary

| Date | Revision History | Revision Class | Comments |
|---|---|---|---|
| 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 | 2.1.0 | Minor | Updated the technical content. |
| 05/05/2010 | 3.0.0 | Major | Updated and revised the technical content. |
| 08/04/2010 | 4.0 | Major | Significantly changed the technical content. |

# Contents

# 1 Introduction

This document specifies the ActiveSync Conversations protocol, which is an **XML**-based format that is used to improve the ways in which e-mail **messages** are triaged when they are displayed in **conversation** view.

## 1.1 Glossary

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

**class**
**conversation**
**conversation ID**
**conversation index**
**Deleted Items folder**
**Hypertext Markup Language (HTML)**
**message**
**message part**
**Wireless Application Protocol (WAP) Binary XML (WBXML)**
**XML**
**XML namespace**
**XML schema**

The following terms are specific to this document:

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

## 1.2 References

### 1.2.1 Normative References

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

[MS-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-ASEMAIL] Microsoft Corporation, "ActiveSync E-Mail Class Protocol Specification", December 2008.

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

[MS-DTYP] Microsoft Corporation, "Windows Data Types", March 2007, http://msdn.microsoft.com/en-us/library/cc230273.aspx

[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] Bray, T., Paoli, J., Sperberg-McQueen, C., Eds., et al., "Extensible Markup Language (XML) 1.0 (Fifth Edition)", W3C Recommendation, November 2008, http://www.w3.org/TR/REC-xml/

[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/

### 1.2.2 Informative References

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

[MS-OXOSFLD] Microsoft Corporation, "Special Folders Protocol Specification", April 2008.

## 1.3 Overview

The ActiveSync Conversations protocol is an XML-based format that is used to improve the ways in which e-mail messages are triaged, allowing a user to view a series of send-response e-mail messages as a single representation, called a conversation.

A conversation appears in a message folder as one unit and allows the user to read the series of related e-mail messages in a single effort. Each e-mail message is assigned a **conversation ID** that is used to identify the conversation to which the e-mail message belongs.

## 1.4 Relationship to Other Protocols

The ActiveSync Conversations protocol consists of a series of XML elements that are embedded inside a command request or a command response. For details about command requests and responses, see [MS-ASCMD]. The **Wireless Application Protocol (WAP) Binary XML (WBXML)**, specified in [MS-ASWBXML], is used to transmit the XML markup that constitutes the request body or the response body.

The ActiveSync Conversations protocol defines elements according to the data type definitions that are specified in [MS-ASDTYPE].

## 1.5 Prerequisites/Preconditions

None.

## 1.6 Applicability Statement

This protocol is applicable in scenarios in which a client needs to synchronize its e-mail message and files with a server and wants to present a view in which e-mail message are grouped by conversation rather than listed serially.

## 1.7 Versioning and Capability Negotiation

None.

## 1.8 Vendor-Extensible Fields

None.

## 1.9 Standards Assignments

None.

# 2 Messages

## 2.1 Transport

The ActiveSync Conversations protocol consists of a series of XML elements that are embedded inside a command request or a command response. The XML markup that constitutes the request body or the response body is transmitted between client and server by using WAP Binary XML (WBXML), as specified in [MS-ASWBXML].

## 2.2 Message Syntax

The XML markup that is used by the ActiveSync Conversations protocol MUST be well-formed XML, as specified in [XML].

The XML elements that are used by the ActiveSync Conversations protocol are embedded inside a request or response for the following commands:

- **GetItemEstimate**
- **ItemOperations**
- **MoveItems**
- **Search**
- **Sync**

For details about the requests and responses for these commands, see [MS-ASCMD].

### 2.2.1 Namespaces

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

| Prefix | Namespace URI | Reference |
|---|---|---|
| airsync | AirSync | [MS-ASCMD] section 2.2.2.19 |
| airsyncbase | AirSyncBase | [MS-ASAIRS] |
| email | Email | [MS-ASEMAIL] |
| email2 | Email2 | [MS-ASEMAIL] |
| getitemestimate | GetItemEstimate | [MS-ASCMD] section 2.2.2.7 |
| itemoperations | ItemOperations | [MS-ASCMD] section 2.2.2.8 |
| search | Search | [MS-ASCMD] section 2.2.2.14 |

### 2.2.2 Elements

The following tables summarize the set of common **XML schema** elements that are defined by this specification for the Email **class**, **GetItemEstimate** command, **ItemOperations** command,

**Search** command, and **Sync** command. For details about the Email class, see [MS-ASEMAIL]. For details about the commands, see [MS-ASCMD] sections 2.2.2.7, 2.2.2.8, 2.2.2.14, and 2.2.2.19.

Elements MUST NOT have child elements in either the command request or command response.

| Element | Description |
|---|---|
| airsyncbase:BodyPart | Used in **ItemOperations**, **Search**, and **Sync** command responses to encapsulate a **message part** and its meta-data. |
| airsyncbase:BodyPartPreference | Used in **ItemOperations**, **Search**, and **Sync** command requests to specify preferences for receiving a message part from the server. |
| email2:ConversationId | Used by the Email class to specify the conversation ID for an e-mail message. |
| email2:ConversationIndex | Used by the Email class to specify the **conversation index** for an e-mail message. |
| airsync:ConversationMode | Used by the **GetItemEstimate** command to enable conversation-based filtering of item estimates. |
| itemoperations:Move | Used in an **ItemOperations** command request to indicate that a conversation is to be moved. |
| itemoperations:ConversationId | Used by the **ItemOperations** command to specify the conversation ID of the conversation that is to be moved. |
| itemoperations:DstFldId | Used by the **ItemOperations** command to specify the destination folder, which is the folder to which the conversation is moved. |
| itemoperations:Options | Used in an **ItemOperations** command request to specify the options for moving a conversation. |
| itemoperations:MoveAlways | Used by the **ItemOperations** command to set up the conversation to be always moved. |
| itemoperations:Move | Used in an **ItemOperations** command response to specify the results of the attempt to move a conversation. |
| itemoperations:Status | Used by the **ItemOperations** command to specify the status of the move action. |
| itemoperations:ConversationId | Used by the **ItemOperations** command to specify the conversation ID of the conversation that is moved. |
| search:ConversationId | Used by the **Search** command to specify the conversation ID of the conversation for which to search. |
| airsync:ConversationMode | Used by the **Sync** command to enable conversation-based filtering and synchronization of conversation-based properties. |

### 2.2.2.1 AirSyncBase Namespace Elements

The following elements are defined in the AirSyncBase namespace. For more details about the AirSyncBase namespace, see [MS-ASAIRS].

#### 2.2.2.1.1 airsyncbase:BodyPart

The <airsyncbase:BodyPart> element<1> encapsulates a message part and its meta-data in a **Sync**, **ItemOperations**, or **Search** command response. The <airsyncbase:BodyPart> element is a child of the <airsync:ApplicationData> element in a **Sync** command response, a child of the <itemoperations:Properties> element in an **ItemOperations** command response, and a child of the <search:Properties> element in a **Search** command response.

The <airsyncbase:BodyPart> element is a **container** element. It has the following child elements:

<airsyncbase:Status>

<airsyncbase:Type>

<airsyncbase:EstimatedDataSize>

<airsyncbase:Truncated>

<airsyncbase:Data>

<airsyncbase:Preview>

The <airsyncbase:BodyPart> element and its child elements are further specified in [MS-ASAIRS]. The **container** data type is specified in [MS-ASDTYPE] section 2.2.

#### 2.2.2.1.2 airsyncbase:BodyPartPreference

The <airsyncbase:BodyPartPreference> element<2> specifies the client's preferences for receiving a message part from the server. The <airsyncbase:BodyPartPreference> element is a child of the <itemoperations:Options>, <search:Options>, or <airsync:Options> element in an **ItemOperations**, **Search**, or **Sync** command request.

The <airsyncbase:BodyPartPreference> element is a **container** element. It has the following child elements:

- <airsyncbase:Type>
- <airsyncbase:TruncationSize>
- <airsyncbase:AllOrNone>
- <airsyncbase:Preview>

The <airsyncbase:BodyPartPreference> element and its child elements are further specified in [MS-ASAIRS]. The **container** data type is specified in [MS-ASDTYPE] section 2.2.

### 2.2.2.2 Email Class Elements

The following elements are defined in the Email2 namespace.

#### 2.2.2.2.1 email2:ConversationId

The <email2:ConversationId> element specifies a unique identifier for a conversation. This element is a required child element of the <airsync:ApplicationData> element in the **Sync** command response, as specified in [MS-ASEMAIL] section 2.2.

The value of this element is a **byte array**, as specified in [MS-ASDTYPE] section 2.6.1. The <email2:ConversationId> element is not present if there is no conversation ID associated with the message.

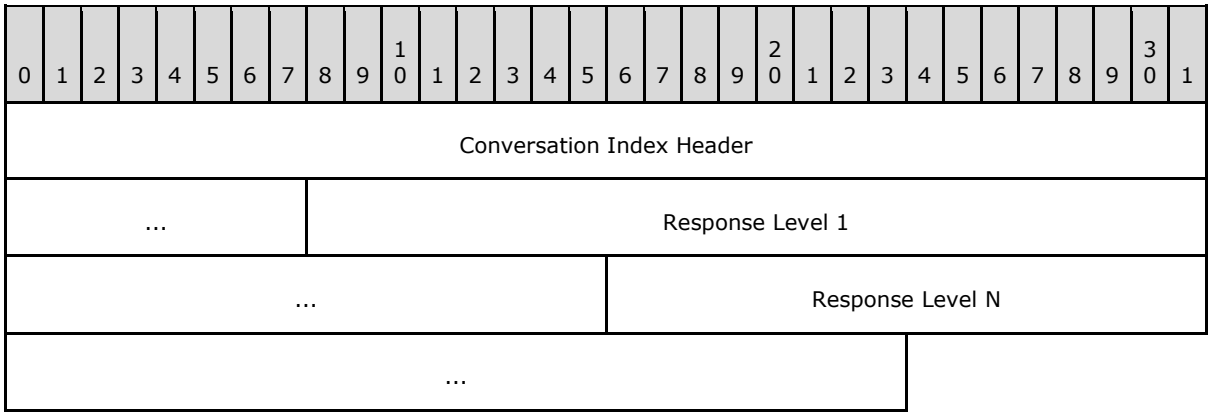
The client MUST NOT change the <email2:ConversationId> value.

#### 2.2.2.2.2 email2:ConversationIndex

The <email2:ConversationIndex> element specifies the conversation index for an e-mail message. This element is a required child element of the <airsync:ApplicationData> element in the **Sync** command response, as specified in [MS-ASEMAIL] section 2.2.

The value of this element is a **byte array**, as specified in [MS-ASDTYPE] section 2.6.1. The value comprises a set of timestamps, which can be used by a client to generate a tree-view of a conversation. The first timestamp identifies the date and time when the message was originally sent by the server. Each additional timestamp specifies the difference between the current time and the time specified by the first timestamp. Additional timestamps are added when the message is forwarded or replied to.

The <email2:ConversationIndex> value is set according to the description in the following diagram.




**Conversation Index Header (5 bytes):** A **Conversation Index Header** that identifies the date and time when the message was originally sent by the server, as specified in section 2.2.2.2.2.1.

**Response Level 1 (5 bytes):** A **Response Level** that contains information about the time the message was forwarded or replied to, as specified in section 2.2.2.2.2.2. Additional **Response Level** fields are added to the <email2:ConversationIndex> each time the message is forwarded or replied to.

**Response Level N (5 bytes):** Additional **Response Level** fields for each time the message is forwarded or replied to. This field is set according to the description for the **Response Level 1** field.

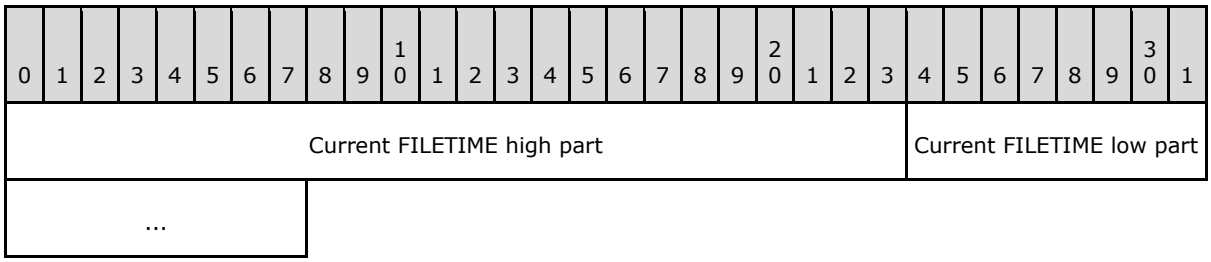
The content of the <email2:ConversationIndex> element is transferred as a **byte array** within the WBXML tags.

The client MUST NOT change the conversation index.

#### 2.2.2.2.2.1 Conversation Index Header

The **Conversation Index Header** identifies the date and time when the message was originally sent by the server.

The **Conversation Index Header** value is set according to the description in the following diagram.




**Current FILETIME high part (3 bytes):** The 24 low bits of the high part of the included **FILETIME** value ([MS-DTYP] section 2.3.1), as shown in the following table.

| **FILETIME expressed in UTC** | | |
|---|---|---|
| Eight most significant bits | 40 bits | 16 least significant bits |
| Excluded | Included | Excluded |

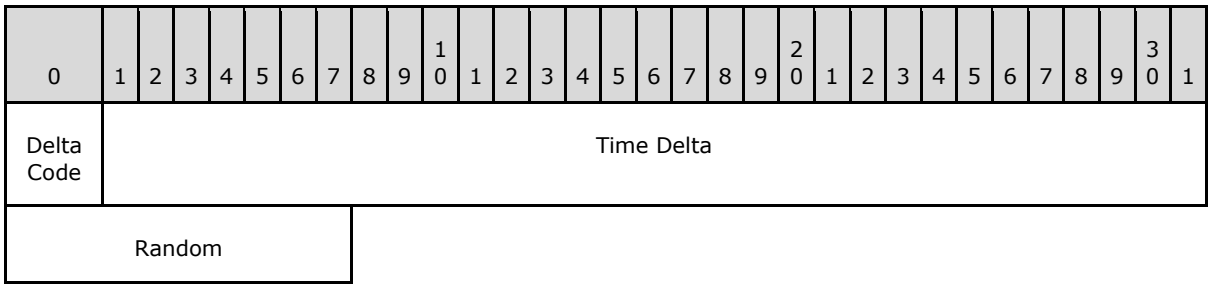
**Current FILETIME low part (2 bytes):** The 16 high bits of the low part of the included **FILETIME** value, as shown in the **Current FILETIME high part** field table.

The data is stored in big-endian format – the five bytes of the time are written from most significant byte to least significant byte.

#### 2.2.2.2.2.2 Response Level

The **Response Level** field contains information about the time the message was forwarded or replied to.

The **Response Level** value is set according to the description in the following diagram.




**Delta Code (1 bit):** A 1-bit flag calculated on the difference between the current time and the time stored in the **Conversation Index Header** (section 2.2.2.2.2.1).

- If the difference is less than 625 days (high order part of the delta **FILETIME** bitwise AND with "0x00FE0000" resulting in "0"), the **Delta Code** is "0".

- If the difference is greater than or equal to 625 days (high order part of the delta **FILETIME** bitwise AND with "0x00FE0000" resulting nonzero), the **Delta Code** is 1.

**Time Delta (31 bits):** A 31-bit value calculated using the difference between the current time and the time stored in the **Conversation Index Header** (section 2.2.2.2.2.1)

- If the difference is less than 625 days (high order part of the delta **FILETIME** bitwise AND with "0x00FE0000" resulting in "0"), the **Time Delta** is the least significant 31 bits of the difference remaining after the 18 least significant bits are excluded.
- If the difference is greater than or equal to 625 days (high order part of the delta **FILETIME** bitwise AND with "0x00FE0000" resulting nonzero), the **Time Delta** is the least significant 31 bits of the difference remaining after the 23 least significant bits are excluded.

**Random (1 byte):** A random value generated by using an implementation-specific algorithm.

### 2.2.2.3 GetItemEstimate Command Elements

The following elements are defined in the **AirSync** namespace.

#### 2.2.2.3.1 airsync:ConversationMode

The <airsync:ConversationMode> element enables or disables conversation-based filtering of item estimates. This element is an optional child element of the <getitemestimate:Collection> element in the **GetItemEstimate** command request. For details about the <getitemestimate:Collection> element, see [MS-ASCMD] section 2.2.2.7.1.1.1.1.

The value of this element is a **Boolean**, as specified in [MS-ASDTYPE] section 2.1. The value 1 enables conversation-based filtering of item estimates; the value zero disables it. If this element is present without a value, the default is 1.

### 2.2.2.4 ItemOperations Command Elements

The following elements are defined in the **ItemOperations** namespace.

#### 2.2.2.4.1 Request

The following elements are used in **ItemOperations** request messages.

#### 2.2.2.4.1.1 itemoperations:Move

The <itemoperations:Move> element indicates that a conversation is to be moved from all folders to a destination folder. The <itemoperations:Move> element is a required **container** element in an **ItemOperations** command request. It contains the following child elements:

- <itemoperations:ConversationId> element — see section 2.2.2.4.1.1.1
- <itemoperations:DstFldId> element — see section 2.2.2.4.1.1.2
- <itemoperations:Options> element — see section 2.2.2.4.1.1.3

The **container** data type is specified in [MS-ASDTYPE] section 2.2.

#### 2.2.2.4.1.1.1 itemoperations:ConversationId

The <itemoperations:ConversationId> element specifies the conversation ID of the conversation that is to be moved. This element is a required child element of the <itemoperations:Move> element in the **ItemOperations** command request.

The value of this element is a **byte array**, as specified in [MS-ASDTYPE] section 2.6.1.

#### 2.2.2.4.1.1.2 itemoperations:DstFldId

The <itemoperations:DstFldId> element specifies the destination folder, which is the folder to which the conversation is moved. This element is a required child element of the <itemoperations:Move> element in the **ItemOperations** command request.

The value of this element is a **string**, as specified in [MS-ASDTYPE] section 2.6. The destination folder MUST be of type "IPF.Note". For more information about folder types, see [MS-OXOSFLD] section 2.2.5.

#### 2.2.2.4.1.1.3 itemoperations:Options

The <itemoperations:Options> element specifies the options for the <itemoperations:Move> element. The <itemoperations:Options> element is an optional **container** element in an **ItemOperations** command request. It contains the following child element:

- <itemoperations:MoveAlways> element — see section 2.2.2.4.1.1.3.1

The **container** data type is specified in [MS-ASDTYPE] section 2.2.

#### 2.2.2.4.1.1.3.1 itemoperations:MoveAlways

The <itemoperations:MoveAlways> element indicates whether a conversation is to be always moved. When a conversation is set to be moved always, all e-mail messages in the conversation, including all future e-mail messages for that conversation, are moved from all folders to a destination folder. This element is an optional child element of the <itemoperations:Options> element in the **ItemOperations** command request.

This element is a flag, which does not have a value. If this element is present, the conversation is set to be moved always.

### 2.2.2.4.2 Response

The following elements are used in **ItemOperations** response messages.

#### 2.2.2.4.2.1 itemoperations:Move

The <itemoperations:Move> element specifies the results of the attempt to move a conversation. The <itemoperations:Move> element is a required **container** element in an **ItemOperations** command response. It contains the following child elements:

- <itemoperations:ConversationId> element — see section 2.2.2.4.2.1.2
- <itemoperations:Status> element — see section 2.2.2.4.2.1.1

The **container** type is specified in [MS-ASDTYPE] section 2.2.

#### 2.2.2.4.2.1.1 itemoperations:Status

The <itemoperations:Status> element specifies the status of the move action. This element is a required child element of the <itemoperations:Move> element in the **ItemOperations** command response.

The value of this element is an **integer**, as specified in [MS-ASDTYPE] section 2.5. The following table lists some of the status values and their meanings.

| Value | Meaning |
|---|---|
| 1 | Success. The server successfully completed the operation. |
| 2 | Protocol error. The XML is not valid. |
| 3 | Server error. There was a complete or partial failure of the operation. |
| 6 | Not Found. The conversation or destination folder does not exist. |
| 105 | Invalid Combination of IDs. The destination folder cannot be the Recipient Information Cache. |
| 155 | Protocol error. The <itemoperations:Options> element does not contain a <itemoperations:MoveAlways> element. |
| 156 | Action not supported. The destination folder MUST be of type "IPF.Note". For more information about folder types, see [MS-OXOSFLD] section 2.2.5. |

#### 2.2.2.4.2.1.2 itemoperations:ConversationId

The <itemoperations:ConversationId> element specifies the conversation ID of the conversation that is moved. This element is a required child element of the <itemoperations:Move> element in the **ItemOperations** command response.

The value of this element is a **byte array**, as specified in [MS-ASDTYPE] section 2.6.1.

### 2.2.2.5 Search Command Elements

The following elements are defined in the **Search** namespace.

#### 2.2.2.5.1 search:ConversationId

The <search:ConversationId> element specifies the conversation ID of the conversation for which to search. This element is an optional child element of the <search:Query> element in the **Search** command request. For details about the <search:Query> element, see [MS-ASCMD] section 2.2.2.14.1.1.1.2.

The value of this element is a **byte array**, as specified in [MS-ASDTYPE] section 2.6.1.

### 2.2.2.6 Sync Command Elements

The following elements are defined in the **AirSync** namespace.

#### 2.2.2.6.1 airsync:ConversationMode

The <airsync:ConversationMode> element enables or disables conversation-based filtering and synchronization of conversation-based properties. This element is an optional child element of the

<airsync:Collection> element in the **Sync** command request. For details about the <airsync:Collection> element, see [MS-ASCMD] section 2.2.2.19.1.2.1.1.

The value of this element is a **Boolean**, as specified in [MS-ASDTYPE] section 2.1. The value 1 enables conversation-based filtering and synchronization of conversation-based properties; the value zero disables it. If this element is present without a value, the default is 1.

# 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.

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

**E-mail messages:** Every e-mail message includes the following to support conversations:

- Conversation ID: A unique value that is associated with a conversation. This value is specified by the <email2:ConversationId> element that is included in the Email class. For details about the Email class, see [MS-ASEMAIL].
- Conversation index: An index that is used by clients to generate a conversation tree view. This value is specified by the <email2:ConversationIndex> element that is included in the Email class. For details about the Email class, see [MS-ASEMAIL].

The server creates a conversation ID and a conversation index on the e-mail item when the user sends an e-mail message. The client does not change the conversation ID or the conversation index.

### 3.1.2 Timers

None.

### 3.1.3 Initialization

None.

### 3.1.4 Higher-Layer Triggered Events

#### 3.1.4.1 Deleting a Conversation

When a conversation is deleted, all e-mail messages that are in the conversation are moved from the current folder to the Deleted Items folder. Future e-mail messages for the same conversation are not affected.

To delete a conversation, the client sends a **Sync** command request that contains a <airsync:Delete> element for each item in the conversation. For more details about the <airsync:Delete> element in the **Sync** command request, see [MS-ASCMD] section 2.2.2.19.1.2.1.1.9.2.

#### 3.1.4.2 Flagging a Conversation for Follow-up

When a conversation is flagged for follow-up, the most recent e-mail message that is in the conversation and that is in the current folder is flagged. Clearing a flag on a conversation will clear flags on all e-mail messages that are in the conversation and that are in the current folder. Marking

a flagged conversation as complete will mark all flagged e-mail messages that are in the conversation and that are in the current folder as complete.

To set a flag on a conversation, clear a flag on a conversation, or mark a flagged conversation as complete, the client sends a **Sync** command request that contains a <airsync:Change> element, as specified in [MS-ASCMD] section 2.2.2.19.1.2.1.1.9.1.

#### 3.1.4.3 Marking a Conversation as Read or Unread

When a conversation is marked as read or unread, all e-mail messages that are in the conversation and that are in the current folder are marked as such.

To mark a conversation as read or unread, the client sends a **Sync** command request that contains a <airsync:Change> element, as specified in [MS-ASCMD] section 2.2.2.19.1.2.1.1.9.1.

#### 3.1.4.4 Ignoring a Conversation

When a conversation is ignored, all e-mail messages in the conversation, including all future e-mail messages for that conversation, are moved from all folders to the Deleted Items folder.

To ignore a conversation, the client sends an **ItemOperations** command request that contains a <itemoperations:Move> element and its child elements, as specified in section 2.2.2.4.1.1 of this document. The <itemoperations:MoveAlways> element MUST be present and the <itemoperations:DstFldId> element MUST contain the ID of the Deleted Items folder. Multiple <itemoperations:Move> elements, one for each conversation to be moved, can be included within one **ItemOperations** request. In this case, the <itemoperations:Move> elements are processed in the order specified. For details about the **ItemOperations** command request, see [MS-ASCMD] section 2.2.2.8.2.

#### 3.1.4.5 Moving a Conversation from the Current Folder

When a conversation is moved from the current folder to another folder, all e-mail messages that are in the conversation are moved from the current folder to the destination folder.

To move a conversation from the current folder to a destination folder, the client sends a **MoveItems** command request, as specified in [MS-ASCMD] section 2.2.2.10.1.

#### 3.1.4.6 Setting up a Conversation to Be Moved Always

When a conversation is set to be moved always, all e-mail messages in the conversation, including all future e-mail messages for that conversation, are moved from all folders to a destination folder.

To set a conversation to be moved always, the client sends an **ItemOperations** command request that contains a <itemoperations:Move> element and its child elements, as specified in section 2.2.2.4.1.1 of this document. The <itemoperations:MoveAlways> element MUST be present. The client MUST NOT specify the Outbox folder, the Drafts folder, or the Recipient Information Cache as the destination folder. For details about the **ItemOperations** command request, see [MS-ASCMD] section 2.2.2.8.2.

#### 3.1.4.7 Finding a Conversation

Searching for a particular conversation will search across all folders for all e-mail messages that are in the conversation.

To search for a conversation, the client sends a **Search** command request with the <search:ConversationId> element, which is specified in section 2.2.2.5.1 of this document. The

<search:ConversationId> element can be used in conjunction with other child elements of the <search:Query> element. The client MUST scope the query to the Email class by setting the <search:Name> element to "Mailbox".

For details about the **Search** command request, see [MS-ASCMD] section 2.2.2.14.1. For details about the <search:Name> element, see [MS-ASCMD] section 2.2.2.14.1.1.1.1. For details about the <search:Query> element and its child elements, see [MS-ASCMD] section 2.2.2.14.1.1.1.2.

#### 3.1.4.8 Synchronizing a Conversation

When a conversation is synchronized, all e-mail messages that are part of the conversation and that are in the specified folder are synchronized.

To synchronize a conversation, the client sends a **Sync** command request with a <airsync:ConversationMode> element for the particular collection to be synchronized.

#### 3.1.4.9 Applying a Conversation-based Filter

Conversation-based filtering augments the date-based filtering. For details about date-based filtering, see [MS-ASCMD] section 2.2.2.19.1.2.1.1.8.1.

When a conversation-based filter is applied to a synchronization of the current folder, the complete conversation is retrieved if any e-mail message in the conversation falls within the date-based filter.

To apply a conversation-based filter to a synchronization, the client includes the <airsync:ConversationMode> element in a **Sync** command request. For details about the **Sync** command request, see [MS-ASCMD] section 2.2.2.19.1.

A conversation-based filter can also be applied to the **GetItemEstimate** command to get an estimate of the items that both meet the filter criteria and need to be synchronized. The client can apply the filter by including the <airsync:ConversationMode> element in a **GetItemEstimate** command request. For details about the **GetItemEstimate** command request, see [MS-ASCMD] section 2.2.2.7.1.

#### 3.1.4.10 Requesting a Message Part

When a client synchronizes, searches, or fetches an e-mail message, the client can choose to receive a message part by including the <airsyncbase:BodyPartPreference> element, which is specified in section 2.2.2.1.2, in the **Sync**, **Search**, or **ItemOperations** command request. The value of the <airsyncbase:Type> element MUST be 2 to specify **Hypertext Markup Language (HTML)** as the encoding format.

### 3.1.5 Message Processing Events and Sequencing Rules

None.

### 3.1.6 Timer Events

None.

### 3.1.7 Other Local Events

None.

## 3.2 Server Details

### 3.2.1 Abstract Data Model

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

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

**E-mail messages:** Every e-mail item includes the following to support conversations:

- Conversation ID: A unique value that is associated with a conversation. This value is specified by the <email2:ConversationId> element that is included in the Email class. For details about the Email class, see [MS-ASEMAIL].
- Conversation Index: An index that is used by clients to generate a conversation tree view. This value is specified by the <email2:ConversationIndex> element that is included in the Email class. For details about the Email class, see [MS-ASEMAIL].

The server creates a conversation ID and a conversation index on the e-mail item when the user sends an e-mail message. The client does not change the conversation ID or the conversation index.

### 3.2.2 Timers

None.

### 3.2.3 Initialization

None.

### 3.2.4 Higher-Layer Triggered Events

The server creates a conversation ID and a conversation index on the e-mail item when the user sends an e-mail message.

### 3.2.5 Message Processing Events and Sequencing Rules

#### 3.2.5.1 Processing a Sync Command

##### 3.2.5.1.1 Deleting a Conversation

The server moves all e-mail messages that are in the conversation from the current folder to the Deleted Items folder. The server does not move future e-mail messages for the conversation.

The server sends a **Sync** response, as specified in [MS-ASCMD] section 2.2.2.19.2.

##### 3.2.5.1.2 Marking a Conversation as Read or Unread

The server marks all e-mails that are in the conversation and that are in the current folder as either read or unread, whichever is specified in the client's request (see section 3.1.4.3).

The server sends a **Sync** response, as specified in [MS-ASCMD] section 2.2.2.19.2.

#### 3.2.5.1.3 Flagging a Conversation for Follow-up

If a conversation is flagged for follow-up, the server flags the most recent e-mail message that is in the conversation and that is in the current folder. If a flag is cleared on a conversation, the server clears flags on all e-mail messages that are in the conversation and that are in the current folder. If a flagged conversation is marked as complete, the server marks all flagged e-mail messages that are in the conversation and that are in the current folder as complete.

The server sends a **Sync** response, as specified in [MS-ASCMD] section 2.2.2.19.2.

### 3.2.5.2 Processing a GetItemEstimate Command

When a conversation-based filter is applied to the **GetItemEstimate** command, the server sends an estimate of the items that meet the filter criteria and need to be synchronized.

In the event of failure, the server sends the following status code. For details about the **GetItemEstimate** response, see [MS-ASCMD] section 2.2.2.7.2.

| Value | Meaning |
|---|---|
| 4 | Protocol error. The conversation-based filter cannot be applied to a folder that is not of the Email class. |

### 3.2.5.3 Processing an ItemOperations Command

#### 3.2.5.3.1 Ignoring a Conversation

When a conversation is ignored, the server moves all e-mail messages in the conversation, including all future e-mail messages for that conversation, from all folders to the Deleted Items folder.

The server's response includes the <itemoperations:Status> element, which contains one of the values specified in section 2.2.2.4.2.1.1, and the <itemoperations:ConversationId> element.

#### 3.2.5.3.2 Always Moving a Conversation

When a conversation is set to be moved always, the server moves all e-mail messages in the conversation, including all future e-mail messages for that conversation, from all folders to a destination folder.

The server's response includes the <itemoperations:Status> element, which contains one of the values specified in section 2.2.2.4.2.1.1, and the <itemoperations:ConversationId> element.

### 3.2.5.4 Processing a MoveItems Command

The server moves all e-mail messages that are in the conversation from the current folder to the destination folder. The server sends a **MoveItems** response, as specified in [MS-ASCMD] section 2.2.2.10.2.

### 3.2.5.5 Processing a Search Command

The server searches across all folders for all e-mail messages that are in the conversation and returns this set of e-mail messages. For details about the **Search** command response, see [MS-ASCMD] section 2.2.2.14.2.

### 3.2.5.6 Filtering

If an individual e-mail message is moved or deleted, and, as a result, the rest of the messages in the conversation fall out of filter, the server SHOULD send soft deletes (**Sync** command response) for those messages only during the aging-off process. The aging-off process is explained in the following paragraph. For details about the **Sync** command and soft deletes, see [MS-ASCMD] sections 2.2.2.19 and 2.2.2.19.2.1.3.1.4.2, respectively.

The aging-off process is the process in which the server deletes objects from the client that are older than the given time-window. The time-window is specified by the client in the <airsync:FilterType> element of the **Sync** command request. For more details about this element, see [MS-ASCMD] section 2.2.2.19.1.2.1.1.8.1. The server typically performs the aging-off process daily at midnight, but the time and frequency of execution is implementation-dependent. An example of how the aging-off process is applied to conversations is as follows: Suppose that the client specifies a three-day time-window. If any e-mail within a conversation is less than three days old, all e-mails (going back in time to the oldest item in the mailbox) within that conversation will be synchronized to the client. Once the newest e-mail within the conversation becomes older than three days, the server will send soft deletes for all of the e-mails that are within the conversation.

### 3.2.5.7 Sending a Message Part

If the client's **Sync**, **Search**, or **ItemOperations** command request includes the <airsyncbase:BodyPartPreference> element, then the server uses the <airsyncbase:BodyPart> element to encapsulate the message part in the response. The <airsyncbase:BodyPart> element is not present in the response when the client did not request the message part, as specified in section 3.1.4.10.

The client's preferences affect the server's response as follows:

- If the size of the message part exceeds the value specified in the <airsyncbase:TruncationSize> element of the request, then the server truncates the message part. The server includes the <airsyncbase:Truncated> element and the <airsyncbase:EstimatedDataSize> element when it truncates the message part.
- If a value other than 2 is specified in the <airsyncbase:Type> element of the request, then the server returns a status value of 164.
- If the client enables conversation-based filtering (section 3.1.4.9), then the server returns a message part for each of the conversation's e-mails that falls outside of the date-based filter.

The <airsyncbase:BodyPart> element and the <airsyncbase:Body> element can co-exist in the response. The presence or absence of <airsyncbase:BodyPart> or <airsyncbase:Body> in the response depends on the preferences specified in the request, as summarized in the following table.

| Request Contains | Response Contains |
|---|---|
| Neither <airsyncbase:BodyPreference> nor <airsyncbase:BodyPartPreference> elements | Only <airsyncbase:Body> element |
| Only <airsyncbase:BodyPreference> element | Only <airsyncbase:Body> element |
| Only <airsyncbase:BodyPartPreference> element | Only <airsyncbase:BodyPart> element |
| Both <airsyncbase:BodyPreference> and <airsyncbase:BodyPartPreference> element | Both <airsyncbase:Body> and <airsyncbase:BodyPart> element |

### 3.2.6 Timer Events

None.

### 3.2.7 Other Local Events

None.

# 4 Protocol Examples

## 4.1 Synchronization From the Server

The following example shows the server returning an e-mail message. Note that the conversation ID and conversation index are included in the <airsync:ApplicationData> node. A server can choose any name for a namespace and then map its chosen name to the actual namespace name. This example shows alternate namespace names being used by the server.

Response from the server:

```
<?xml version="1.0" encoding="utf-8"?>
<Sync xmlns:email="Email:" xmlns:airsyncbase="AirSyncBase:"
xmlns:email2="Email2:" xmlns="AirSync:">
  <Collections>
    <Collection>
      <SyncKey>1601897837</SyncKey>
      <CollectionId>7</CollectionId>
      <Status>1</Status>
      <Commands>
        <Add>
          <ServerId>7:1</ServerId>
          <ApplicationData>
            <email:To>"deviceuser" &lt;someone@example.com&gt;</email:To>
            <email:From>"deviceuser2" &lt;someone2@example.com&gt;</email:From>
            <email:Subject>Test report</email:Subject>
            <email:DateReceived>2009-03-21T07:04:26.948Z</email:DateReceived>
            <email:DisplayTo>deviceuser</email:DisplayTo>
            <email:ThreadTopic>Test report</email:ThreadTopic>
            <email:Importance>1</email:Importance>
            <email:Read>1</email:Read>
            <airsyncbase:Body>
              <airsyncbase:Type>1</airsyncbase:Type>
              <airsyncbase:EstimatedDataSize>100</airsyncbase:EstimatedDataSize>
              <airsyncbase:Truncated>1</airsyncbase:Truncated>
              <airsyncbase:Data>Test data</airsyncbase:Data>
            </airsyncbase:Body>
            <email:MessageClass>IPM.Note</email:MessageClass>
            <email:InternetCPID>20127</email:InternetCPID>
            <email:Flag/>
            <email:ContentClass>urn:content-classes:message</email:ContentClass>
            <airsyncbase:NativeBodyType>2</airsyncbase:NativeBodyType>
            <email2:ConversationId>BBA4726D4399D44C83297D4BD904ED2D</email2:ConversationId>
            <email2:ConversationIndex>01C9A9F345</email2:ConversationIndex>
            <email:Categories/>
          </ApplicationData>
        </Add>
      </Commands>
    </Collection>
  </Collections>
</Sync>
```

## 4.2 Ignoring a Conversation

The following example shows the client's request to ignore a conversation and the server's response.

Request from the client:

```
<ItemOperations xmlns="ItemOperations:">
  <Move>
    <ConversationId>...</ConversationId>
    <DstFldId>DeletedItems-Folder-ID</DstFldId>
    <Options>
      <MoveAlways/>
    </Options>
  </Move>
</ItemOperations>
```

Response from the server:

```
<ItemOperations xmlns="ItemOperations:">
  <Response>
    <Status>1</Status>
    <Move>
      <Status>1</Status>
      <ConversationId>...</ConversationId>
    </Move>
  </Response>
</ItemOperations>
```

# 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
- Microsoft® Exchange Server 2010 Service Pack 1 (SP1)

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.

<1> Section 2.2.2.1.1: The <airsyncbase:BodyPart> element is not supported when the MS-ASProtocolVersion header is set to 12.1 or 14.0.

<2> Section 2.2.2.1.2: The <airsyncbase:BodyPartPreference> element is not supported when the MS-ASProtocolVersion header is set to 12.1 or 14.0.

# 7 Change Tracking

This section identifies changes that were made to the [MS-ASCON] protocol document between the May 2010 and August 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.
- 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 |
|---|---|---|---|
| 1.1 Glossary | 55149<br>Added "XML namespace" to the list of terms defined in [MS-OXGLOS]. | N | Content update. |
| 1.1 Glossary | 55911<br>Changed the term "WAP Binary XML (WBXML)" to "Wireless Application Protocol (WAP) Binary XML (WBXML)" in the list of terms defined in [MS-OXGLOS]. | N | Content update. |
| 1.1 Glossary | 56900<br>Removed the term "GUID" from the list of terms defined in [MS-OXGLOS]. | N | Content update. |
| 1.1 Glossary | 56621<br>Added “Deleted Items folder” to the list of terms defined [MS-OXGLOS]. | N | Content update. |
| 1.2.1 Normative References | 55751<br>Moved [MS-OXGLOS] from Normative References section to Informative References section. | N | Content update. |
| 1.2.1 Normative References | 55149<br>Added [XMLNS] as a normative reference. | N | Content update. |
| 1.2.1 | 57648 | N | Content update. |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| Normative References | Added [MS-ASAIRS] as a normative reference. | | |
| 1.2.2 Informative References | 57648 Added [MS-OXOSFLD] as an informative reference. | N | Content update. |
| 1.4 Relationship to Other Protocols | 55911 Changed "WAP Binary XML (WBXML)" to "Wireless Application Protocol (WAP) Binary XML (WBXML)". | N | Content update. |
| 1.4 Relationship to Other Protocols | 55560 Removed references to complex types. | N | Content update. |
| 2.2 Message Syntax | 55149 Removed the list of namespaces, which are now defined in the Namespaces section. | N | Content removed. |
| 2.2.1 Namespaces | 55149 Added new section. | Y | New content added for template compliance. |
| 2.2.2 Elements | 55149 Added namespace prefixes to all elements. | N | New content added for template compliance. |
| 2.2.2.1 AirSyncBase Namespace Elements | 55560 Changed the title and text to reflect that this section discusses elements, not complex types. | N | Content update. |
| 2.2.2.1.1 airsyncbase:BodyPart | 55560 Updated text to refer to the element as an element and not a complex type. | N | Content update. |
| 2.2.2.1.1 airsyncbase:BodyPart | 55149 Added namespace prefixes to all elements. | N | New content added for template compliance. |
| 2.2.2.1.1 airsyncbase:BodyPart | 56010 Added a product behavior note stating which MS-ASProtocolVersion values are supported. | N | New product behavior note added. |
| 2.2.2.1.2 airsyncbase:BodyPartPreference | 55560 Updated the text to refer to the element as an element and not a complex type. | N | Content update. |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| 2.2.2.1.2 airsyncbase:BodyPartPreference | 55149<br>Added namespace prefixes to the title and text. | N | New content added for template compliance. |
| 2.2.2.1.2 airsyncbase:BodyPartPreference | 56010<br>Added a product behavior note stating which MS-ASProtocolVersion values are supported. | N | New product behavior note added. |
| 2.2.2.2.1 email2:ConversationId | 55560<br>Updated the text to refer to the elements as elements and not complex types. | N | Content update. |
| 2.2.2.2.1 email2:ConversationId | 55149<br>Added namespace prefixes to the title and text. | N | New content added for template compliance. |
| 2.2.2.2.2 email2:ConversationIndex | 55560<br>Updated the text to refer to elements as elements and not complex types. | N | Content update. |
| 2.2.2.2.2 email2:ConversationIndex | 55149<br>Added namespace prefixes to the title and text. | N | New content added for template compliance. |
| 2.2.2.2.2 email2:ConversationIndex | 54593<br>Added bitmap diagrams and field descriptions to specify the element. | Y | Content update. |
| 2.2.2.2.2.1 Conversation Index Header | 54593<br>Added new section. | N | New content added. |
| 2.2.2.2.2.2 Response Level | 54593<br>Added new section. | N | New content added. |
| 2.2.2.3.1 airsync:ConversationMode | 55560<br>Updated the text to refer to elements as elements and not complex types. | N | Content update. |
| 2.2.2.3.1 airsync:ConversationMode | 55149<br>Added namespace prefixes to and removed dot notation from the title and text. | N | New content added for template compliance. |
| 2.2.2.4.1 Request | 55560<br>Added new section. | N | New content added. |
| 2.2.2.4.1.1 itemoperations:Move | 55560<br>Updated the text to refer to the element as an element and not a complex type and removed usage of | N | Content update. |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| | dot notation in element name. | | |
| 2.2.2.4.1.1 itemoperations:Move | 55149<br>Added namespace prefixes to the title and text. | N | New content added for template compliance. |
| 2.2.2.4.1.1.1 itemoperations:ConversationId | 55560<br>Updated the text to refer to the elements as elements and not complex types. | N | Content update. |
| 2.2.2.4.1.1.1 itemoperations:ConversationId | 55149<br>Added namespace prefixes to the title and text. Removed dot notation from title. | N | New content added for template compliance. |
| 2.2.2.4.1.1.2 itemoperations:DstFldId | 55560<br>Updated the text to elements as elements and not complex types. | N | Content update. |
| 2.2.2.4.1.1.2 itemoperations:DstFldId | 55149<br>Added namespace prefixes to the title and text Removed dot notation from title. | N | New content added for template compliance. |
| 2.2.2.4.1.1.3 itemoperations:Options | 55560<br>Updated the text to refer to the element as an element and not a complex type and removed the use of dot notation in the element name. | N | Content update. |
| 2.2.2.4.1.1.3 itemoperations:Options | 55149<br>Added namespace prefixes to the title and text. | N | New content added for template compliance. |
| 2.2.2.4.1.1.3.1 itemoperations:MoveAlways | 55560<br>Updated the text to refer to elements as elements and not complex types. Removed dot notation from element names. | N | Content update. |
| 2.2.2.4.1.1.3.1 itemoperations:MoveAlways | 55149<br>Added namespace prefixes to the title and text. | N | New content added for template compliance. |
| 2.2.2.4.2 Response | 55560<br>Added new section. | N | New content added. |
| 2.2.2.4.2.1 itemoperations:Move | 55560<br>Updated the text to refer to the element as an element and not a complex type and removed the use of | N | Content update. |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| | dot notation in the element name. | | |
| 2.2.2.4.2.1 itemoperations:Move | 55149<br>Added namespace prefixes to the title and text. | N | New content added for template compliance. |
| 2.2.2.4.2.1.1 itemoperations:Status | 55560<br>Updated the text to refer to elements as elements and not complex types. | N | Content update. |
| 2.2.2.4.2.1.1 itemoperations:Status | 55149<br>Added namespace prefixes to the title and text. | N | New content added for template compliance. |
| 2.2.2.4.2.1.2 itemoperations:ConversationId | 55560<br>Updated the text to refer to elements as elements and not complex types. | N | Content update. |
| 2.2.2.4.2.1.2 itemoperations:ConversationId | 55149<br>Added namespace prefixes to the title and text. | N | New content added for template compliance. |
| 2.2.2.5.1 search:ConversationId | 55560<br>Updated the text to refer to elements as elements and not complex types. | N | Content update. |
| 2.2.2.5.1 search:ConversationId | 55149<br>Added namespace prefixes to and removed dot notation from the title and text. | N | New content added for template compliance. |
| 2.2.2.6.1 airsync:ConversationMode | 55560<br>Updated the text to refer to elements as elements and not as complex types. | N | Content update. |
| 2.2.2.6.1 airsync:ConversationMode | 55149<br>Added namespace prefixes to and removed dot notation from the title and text. | N | New content added for template compliance. |
| 3.1.1 Abstract Data Model | 55149<br>Added namespace prefixes to all elements. | N | Content update. |
| 3.1.4.1 Deleting a Conversation | 55149<br>Added namespace prefixes to all elements. | N | New content added for template compliance. |
| 3.1.4.2 Flagging a Conversation for | 55149<br>Added namespace prefixes to all | N | New content added for |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| Follow-up | elements. | | template compliance. |
| 3.1.4.3 Marking a Conversation as Read or Unread | 55149<br>Added namespace prefixes to all elements. | N | New content added for template compliance. |
| 3.1.4.4 Ignoring a Conversation | 55560<br>Updated the text to refer to the elements as elements and not complex types. | N | Content update. |
| 3.1.4.4 Ignoring a Conversation | 55149<br>Added namespace prefixes to and removed dot notation from all elements. | N | New content added for template compliance. |
| 3.1.4.6 Setting up a Conversation to Be Moved Always | 55149<br>Added namespace prefixes to and removed dot notation from all elements. | N | New content added for template compliance. |
| 3.1.4.6 Setting up a Conversation to Be Moved Always | 55560<br>Updated the text to refer to the elements as elements and not complex types. | N | Content update. |
| 3.1.4.7 Finding a Conversation | 55560<br>Updated the text to refer to the elements as elements and not complex types. | N | Content update. |
| 3.1.4.7 Finding a Conversation | 55149<br>Added namespace prefixes to and removed dot notation from all elements. | N | New content added for template compliance. |
| 3.1.4.8 Synchronizing a Conversation | 55149<br>Added namespace prefixes to all elements. Removed dot notation from element name. | N | New content added for template compliance. |
| 3.1.4.9 Applying a Conversation-based Filter | 55149<br>Added namespace prefixes to all elements. Removed dot notation from element names. | N | New content added for template compliance. |
| 3.1.4.10 Requesting a Message Part | 55149<br>Added namespace prefixes to all elements. | N | New content added for template compliance. |
| 3.1.4.10 Requesting a Message Part | 55560<br>Updated the text to refer to the | N | Content update. |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| | elements as elements and not complex types. | | |
| 3.2.1 Abstract Data Model | 55149 Added namespace prefixes to all elements. | N | New content added for template compliance. |
| 3.2.5.3.1 Ignoring a Conversation | 55149 Added namespace prefixes to all elements. Removed dot notation from element names. | N | New content added for template compliance. |
| 3.2.5.3.2 Always Moving a Conversation | 55149 Added namespace prefixes to all elements. Removed dot notation from element names. | N | New content added for template compliance. |
| 3.2.5.6 Filtering | 55149 Added namespace prefixes to all elements. | N | New content added for template compliance. |
| 3.2.5.7 Sending a Message Part | 55560 Updated the text to refer to the elements as elements and not complex types. | N | Content update. |
| 3.2.5.7 Sending a Message Part | 55149 Added namespace prefixes to all elements. | N | New content added for template compliance. |
| 4.1 Synchronization From the Server | 55149 Updated and added namespace prefixes and removed product behavior note about namespace prefixes. | N | New content added for template compliance. |
| 4.2 Ignoring a Conversation | 55149 Added the default namespace definition to the example. | N | New content added for template compliance. |
| 6 Appendix A: Product Behavior | 56663 Removed "Beta" from Exchange Server 2010 SP1 product name. | N | Content update. |
| 2.2.1 Complex Types | 55560 Removed the Complex Types section. | N | Content removed. |

# 8 Index