

[MS-ASAIRS]: ActiveSync AirSyncBase Namespace 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.

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

Revision Summary

Date	Revision History	Revision Class	Comments
12/03/2008	1.0	Major	Initial Release.
03/04/2009	1.0.1	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	3.0.2	Editorial	Updated the technical content.
05/05/2010	4.0.0	Major	Updated and revised the technical content.

Table of 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	7
2 Messages	8
2.1 Transport	8
2.2 Message Syntax	8
2.2.1 Complex Types	10
2.2.1.1 Attachments	10
2.2.1.2 Attachment	10
2.2.1.3 Body	11
2.2.1.4 BodyPreference	11
2.2.1.5 BodyPart	12
2.2.1.6 BodyPartPreference	13
2.2.2 Elements	13
2.2.2.1 AllOrNone	14
2.2.2.2 ContentId	15
2.2.2.3 ContentLocation	15
2.2.2.4 Data	15
2.2.2.5 DisplayName	15
2.2.2.6 EstimatedDataSize	16
2.2.2.7 FileReference	16
2.2.2.8 IsInline	16
2.2.2.9 Method	17
2.2.2.10 NativeBodyType	17
2.2.2.11 Preview	18
2.2.2.12 Status	18
2.2.2.13 Truncated	19
2.2.2.14 TruncationSize	19
2.2.2.15 Type	19
2.2.3 Attributes	20
2.2.4 Groups	20
2.2.4.1 TopLevelSchemaProps	20
2.2.5 Attribute Groups	20
3 Protocol Details	21
3.1 Client Details	21
3.1.1 Abstract Data Model	21
3.1.2 Timers	21
3.1.3 Initialization	21
3.1.4 Higher-Layer Triggered Events	21

3.1.5	Message Processing Events and Sequencing Rules	21
3.1.5.1	Commands	21
3.1.5.1.1	Fetch	21
3.1.5.1.2	Search	21
3.1.5.1.3	Sync	21
3.1.6	Timer Events	22
3.1.7	Other Local Events	22
3.2	Server Details	22
3.2.1	Abstract Data Model	22
3.2.2	Timers	22
3.2.3	Initialization	22
3.2.4	Higher-Layer Triggered Events	22
3.2.5	Message Processing Events and Sequencing Rules	22
3.2.5.1	Commands	22
3.2.5.1.1	Sync	22
4	Protocol Examples	23
5	Security	24
5.1	Security Considerations for Implementers	24
5.2	Index of Security Parameters	24
6	Appendix A: Product Behavior	25
7	Change Tracking	26
8	Index	35

1 Introduction

This document specifies the elements and complex types in the AirSyncBase namespace, which are used by the AirSync commands specified in [MS-ASCMD] to identify the size, type, and content of the data returned to the client in the response message. The AirSyncBase namespace contains elements and complex types used in both request and response command messages.

1.1 Glossary

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

attachment
attachments table
calendar
class
collection
contact
Embedded Message object
Hypertext Markup Language (HTML)
identifier
message
message body
message part
MIME
Rich Text Format (RTF)
plain text
property
rules
store
Unicode
Uniform Resource Locator (URL)
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-ASCMD] Microsoft Corporation, "[ActiveSync Command Reference Protocol Specification](#)", December 2008.

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

[MS-AEMAIL] Microsoft Corporation, "[ActiveSync E-Mail Class Protocol Specification](#)", December 2008.

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

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, BCP 14, March 1997, <http://www.ietf.org/rfc/rfc2119.txt>

[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

None.

1.3 Overview

The elements and complex types specified in the AirSyncBase namespace are used by multiple Exchange ActiveSync commands to identify the size, type, and content of data returned to the client in the response message. In order to use the elements in the AirSyncBase namespace, the namespace is included in the command response message, and then elements and complex types are included in the request and response as specified in this document.

1.4 Relationship to Other Protocols

The **ItemOperations** ([\[MS-ASCMD\]](#) section 2.2.1.8), **Search** ([\[MS-ASCMD\]](#) section 2.2.1.14), and **Sync** ([\[MS-ASCMD\]](#) section 2.2.1.19) commands use the ActiveSync E-mail, **Contact**, Note, and Task **classes** in their request and response syntax. These classes use types and elements from the AirSyncBase namespace, as specified in [\[MS-AEMAIL\]](#), [\[MS-ASCNTC\]](#), [\[MS-ASNOTE\]](#), and [\[MS-ASTASK\]](#).

The types and elements in this specification use data types specified in [\[MS-ASDTYPE\]](#).

1.5 Prerequisites/Preconditions

To use the **properties** in the AirSyncBase namespace, include the namespace in the command request. The namespace is included by adding the following to the command request:

```
<CommandName xmlns:airsyncbase="ClassName:">
```

For a complete example, see [\[MS-ASCMD\]](#) section 4.7.1.1.

1.6 Applicability Statement

This specification applies to the **Fetch**, **Search**, and **Sync** commands, as specified in [\[MS-ASCMD\]](#).

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

The <Type> element can be extended to include custom **message** types. For more details, see section [2.2.2.15](#).

1.9 Standards Assignments

None.

PRELIMINARY

2 Messages

2.1 Transport

The properties specified in the following sections are sent and received by using the **Fetch**, **Search**, and **Sync** commands, as specified in [\[MS-ASCMD\]](#), and are used by the E-mail **class**, as specified in [\[MS-ASEMAIL\]](#).

2.2 Message Syntax

The AirSyncBase namespace adheres to the following **XML schema definition (XSD)**:

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

  <xs:element name="FileReference" type="xs:string" />
  <xs:element name="BodyPreference">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Type" type="xs:unsignedByte" />
        <xs:element name="TruncationSize" minOccurs="0" type="xs:unsignedInt" />
        <xs:element name="AllOrNone" minOccurs="0" type="xs:boolean"/>
        <xs:element name="Preview" minOccurs="0">
          <xs:simpleType>
            <xs:restriction base="xs:unsignedInt">
              <xs:minInclusive value="0"/>
              <xs:maxInclusive value="255"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="BodyPartPreference">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Type">
          <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
              <xs:minInclusive value="1"/>
              <xs:maxInclusive value="4"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
        <xs:element name="TruncationSize" minOccurs="0" type="xs:unsignedInt" />
        <xs:element name="AllOrNone" minOccurs="0" type="xs:boolean"/>
        <xs:element name="Preview" minOccurs="0">
          <xs:simpleType>
            <xs:restriction base="xs:unsignedInt">
              <xs:minInclusive value="0"/>
              <xs:maxInclusive value="255"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```



```

        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Body">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Type" type="xs:unsignedByte" />
      <xs:element name="EstimatedDataSize" type="xs:unsignedInt" minOccurs="0"/>
      <xs:element name="Truncated" minOccurs="0" type="xs:boolean"/>
      <xs:element name="Data" minOccurs="0" type="xs:string" />
      <xs:element name="Preview" minOccurs="0" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="BodyPart">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Status" type="xs:unsignedByte" />
      <xs:element name="Type" type="xs:unsignedByte" />
      <xs:element name="EstimatedDataSize" type="xs:unsignedInt" />
      <xs:element name="Truncated" minOccurs="0" type="xs:boolean"/>
      <xs:element name="Data" minOccurs="0" type="xs:string" />
      <xs:element name="Preview" minOccurs="0" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Attachments">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Attachment">
        <xs:complexType>
          <xs:all>
            <xs:element name="DisplayName" type="xs:string" minOccurs="0"/>
            <xs:element name="FileReference" type="xs:string" />
            <xs:element name="Method" type="xs:unsignedByte" />
            <xs:element name="EstimatedDataSize" type="xs:unsignedInt" />
            <xs:element name="ContentId" type="xs:string" minOccurs="0" />
            <xs:element name="ContentLocation" type="xs:string"
minOccurs="0"/>
            <xs:element name="IsInline" minOccurs="0" type="xs:boolean"/>
          </xs:all>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:complexType name="EmptyTag"/>
<xs:element name="NativeBodyType" type="xs:unsignedByte" />
<xs:group name="TopLevelSchemaProps">
  <xs:sequence>
    <xs:choice maxOccurs="unbounded">
      <xs:element name="Body" type="airsyncbase:EmptyTag"/>
      <xs:element name="BodyPart" type="airsyncbase:EmptyTag"/>
      <xs:element name="Attachments" type="airsyncbase:EmptyTag"/>
    </xs:choice>
  </xs:sequence>

```

```

    </xs:group>
</xs:schema>

```

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

Prefix	Namespace URI
AIRSYNCBASE:	None.

2.2.1 Complex Types

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

Complex Type	Description
Attachments	A collection of elements that contains one or more attachment items.
Attachment	Specifies the attachment information for a single attachment item.
Body	A collection of elements that specifies a free-form, variable length data field associated with a stored item on the server.
BodyPreference	A collection of elements that sets the preference information related to the type and size of information that is returned from searching, synchronizing, or fetching.
BodyPart	A collection of elements that contains the message part of the body of an e-mail.
BodyPartPreference	A collection of elements that sets the preference information related to the type and size of information that is returned from searching, synchronizing, or fetching a BodyPart .

2.2.1.1 Attachments

The **Attachments** type is an optional child element of the <Properties> element ([\[MS-ASCMD\]](#) section 2.2.1.8.3.10) in the **Fetch** command request that contains one or more attachment items.

There is no limit on the number of **Attachments** types in a command response. Command requests **MUST NOT** include the **Attachments** type. If a command request includes the **Attachments** type, then the server responds with a protocol status error of 2.

The **Attachments** type, if present, **MUST** have at least one child **Attachment** type.

2.2.1.2 Attachment

The **Attachment** type is a required child element of the **Attachments** type and specifies the attachment information for a single attachment item.

Command requests MUST NOT include the **Attachment** type. If a command request includes the **Attachment** type, the server responds with a protocol status error of 2.

The **Attachment** type can have one or more of the following child elements in any order (elements noted as optional can be in the response):

- <DisplayName> (section [2.2.2.5](#)). This element is optional.
- <FileReference> (section [2.2.2.7](#)). This element is required.
- <Method> (section [2.2.2.9](#)). This element is required.
- <EstimatedDataSize> (section [2.2.2.6](#)). This element is required.
- <ContentId> (section [2.2.2.2](#)). This element is optional.
- <ContentLocation> (section [2.2.2.3](#)). This element is optional.
- <IsInline> (section [2.2.2.8](#)). This element is optional.

2.2.1.3 Body

The **Body** type is an optional child type of the <ApplicationData> element that specifies a free-form, variable length data field associated with a stored item on the server. Examples include, but are not limited to, the body of an e-mail message and the Notes field in a contact. The **Body** type MUST be included in a response message whenever a content class has changes or new items.

The **Body** type is a container ([\[MS-ASDTYPE\]](#) section 2.8).

There is no limit on the number of **Body** types in a command response. When included in a command response, the **Body** type indicates the existence of one or more variable-length fields of data associated with the item. Command requests can include the **Body** type.

In a response, the <ApplicationData> element MUST be the parent element of the **Body** type. For more details about the <ApplicationData> element, see [\[MS-ASCMD\]](#) section 2.2.1.19.2.2.

The **Body** type, if present, has the following required and optional child elements in this order:

- <Type> (section [2.2.2.15](#)). This element is required.
- <Preview> (section [2.2.2.11](#)). This element is optional.
- <EstimatedDataSize> (section [2.2.2.6](#)). This element is optional.
- <Truncated> (section [2.2.2.13](#)). This element has no effect in a command request and is optional in the response.
- <Data> (section [2.2.2.4](#)). This element is optional.

2.2.1.4 BodyPreference

The **BodyPreference** type is an optional container ([\[MS-ASDTYPE\]](#) section 2.8) complex type that sets preference information related to the type and size of information that is returned from searching, synchronizing, or fetching.

A command response MUST NOT include the **BodyPreference** type. Command requests can include the **BodyPreference** type.

In a request, the <Options> element MUST be the parent element of the **BodyPreference** type. If it is not, then the server returns a protocol status error. The **BodyPreference** type, if present, can have the following child elements in this order:

- <Type> (section [2.2.2.15](#)). This element is required.
- <TruncationSize> (section [2.2.2.14](#)). This element is optional.
- <AllOrNone> (section [2.2.2.1](#)). This element is optional.
- <Preview> (section [2.2.2.11](#)). This element is optional.

If the client issues a **Sync** command request with these elements in a different order, then the server responds with a protocol status error of 4. If the client issues an **ItemOperations** command request with these elements in a different order, then the server responds with a protocol status error of 2.

The contents of the <Options> element specify preferences for all of the content that the user is interested in searching, synchronizing, or retrieving. These preferences are persisted by the server from request to request for the specified client, and can be changed by the inclusion of an <Options> element in any subsequent request.

The client specifies the sets of preferences for different types of content by using a separate **BodyPreference** type for each <Type>. If multiple content types have the same preferences, then the client sends a different complex type for each <Type> to prevent ambiguity in specifying the preferences.

The client MUST NOT provide more than one **BodyPreference** type for each allowable value of the <Type> element. If the client includes two or more **BodyPreference** types with the same value of the <Type> element in a **Sync** command request, then the server responds with a protocol status error of 4; if the client includes such values in an **ItemOperations** command request, then the server responds with a protocol status error of 2.

2.2.1.5 BodyPart

The **BodyPart** type is an optional child type of the <ApplicationData> element that specifies details about the message part of an e-mail in a response. The **BodyPart** type MUST be included in a command response when the **BodyPartPreference** type is specified in a request.

The **BodyPart** type is a container ([\[MS-ASDTYPE\]](#) section 2.8).

There is no limit on the number of **BodyPart** types in a command response. Command requests MUST NOT include the **BodyPart** type.

In a response, the <ApplicationData> element MUST be the parent element of the **BodyPart** type. For more information details about the <ApplicationData> element, see [\[MS-ASCMD\]](#) section 2.2.1.19.1.7.

The **BodyPart** type, if present, MUST have its required and optional child elements in the following order:

- <Status> (section [2.2.2.12](#)). This element is required.
- <Type> (section [2.2.2.15](#)). This element is required.
- <EstimatedDataSize> (section [2.2.2.6](#)). This element is required.

- <Truncated> (section [2.2.2.13](#)). This element is optional.
- <Data> (section [2.2.2.4](#)). This element is optional.
- <Preview> (section [2.2.2.11](#)). This element is optional.

2.2.1.6 BodyPartPreference

The **BodyPartPreference** type is an optional container ([\[MS-ASDTYPE\]](#) section 2.8) complex type that sets preference information related to the type and size of information that is returned from searching, synchronizing, or fetching a message part.

A command response MUST NOT include a **BodyPartPreference** type. Command requests can include the **BodyPartPreference** type.

In a request, the <Options> element MUST be the parent element of the **BodyPartPreference** type. The **BodyPartPreference** type, if present, MUST have the following required and optional child elements in the following order:

- <Type> (section [2.2.2.15](#)). This element is required.
- <TruncationSize> (section [2.2.2.14](#)). This element is optional.
- <AllOrNone> (section [2.2.2.1](#)). This element is optional.
- <Preview> (section [2.2.2.11](#)). This element is optional.

The contents of the <Options> element specify preferences for all of the content that the user is interested in searching, synchronizing, or retrieving. These preferences are set on a per-request basis and override any stored information. Because this information is required to process every request, the information can be persisted on the server if network load is a concern.

There MUST be one explicit **BodyPartPreference** for each <Type> value specified in the set of preferences in order to request a **BodyPart** of that <Type> in the response. The server SHOULD [<1>](#) support a <Type> value of 2 (HTML) in the response and MAY support other values. The client MUST specify the **BodyPartPreference** type when a **BodyPart** is requested as part of the server response. The server only returns a **BodyPart** in the response if a **BodyPartPreference** type has been received in a client request.

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
AllOrNone	Specifies whether to search, synchronize, or retrieve all or none of the content based on the <TruncationSize> element.
ContentId	Contains the unique object ID for an attachment.
ContentLocation	Contains the relative URL for an attachment, which is used to match a reference to an inline attachment in an HTML message to the attachment in the attachments table .
Data	The body of the calendar item, contact, document, e-mail, or task.

Element	Description
DisplayName	The display name of the attachment.
EstimatedDataSize	An informational estimate of the size of the data associated with the item's Body or BodyPart type.
FileReference	The location of an item on the server to retrieve.
IsInline	Specifies whether the attachment is embedded in the message.
Method	Identifies the method in which the attachment was attached.
NativeBodyType	The original format type of the item.
Preview	The length of the message preview to be returned to the client.
Status	The status of the <Data> element within the BodyPart response.
Truncated	Specifies whether the body of the item has been truncated according to the BodyPreference or BodyPartPreference indicated by the client.
TruncationSize	Specifies the size, in bytes, of the content that the client wants to search, synchronize, or fetch.
Type	The format type of the body content of the item.

2.2.2.1 AllOrNone

The <AllOrNone> element is an optional child element of the **BodyPreference** type and the **BodyPartPreference** type that specifies whether to search, synchronize, or retrieve all or none of the content based on the <TruncationSize> (section [2.2.2.14](#)) element.

The value of this element is a **boolean** value ([\[MS-ASDTYPE\]](#) section 2.3). If a non-boolean value is included in a client request, then the server responds with a protocol status error of 4. When the value is set to 1 (TRUE) and the content has not been truncated, all of the content is searched, synchronized, or retrieved. When the value is set to 1 (TRUE) and the content has been truncated, the content is not searched, synchronized, or retrieved. When the value is set to 0 (FALSE), the truncated or non-truncated content is searched, synchronized, or retrieved.

A command request **MUST** have a maximum of 1 <AllOrNone> value per **BodyPreference** or **BodyPartPreference** container element. If a command request includes more than one <AllOrNone> value per **BodyPreference** or **BodyPartPreference** container element, then the server returns a protocol status error of 4. If the <AllOrNone> value is not included in the request, then the truncated or non-truncated content is searched, synchronized, or retrieved as if the value was set to 0 (FALSE). The <AllOrNone> element **MUST NOT** be used in command responses.

This element **MUST** be ignored if the <TruncationSize> element is not included.

A client can include multiple **BodyPreference** or **BodyPartPreference** blocks in a command request with different values for the <Type> element (section [2.2.2.15](#)). By default, the server returns the data truncated to the size requested by <TruncationSize> for the <Type> element that matches the native storage format of the item's <Body>. But, if the client also includes the <AllOrNone> element along with the <TruncationSize> element, it is instructing the server not to return a truncated response for that type when the size (in bytes) of the available data exceeds the value of the <TruncationSize> element. For example, a client can use these two elements to signify that it cannot process partial **Rich Text Format (RTF)** data (a <Type> element value of 3). In this

case, if the client has specified multiple **BodyPreference** or **BodyPartPreference** blocks, then the server chooses one of the client's other preferred types.

2.2.2.2 ContentId

The <ContentId> element is an optional child element of the **Attachment** type that contains the unique object ID for an attachment. This element is provided for informational purposes only and can be ignored by the client.

The value of this element is a **string** value ([MS-ASDTYPE] section 2.1).

A command response MUST have a maximum of one <ContentId> element per **Attachment** type. Command requests MUST NOT include the <ContentId> element. If a command request includes the <ContentId> element, the server returns a protocol status error.

The <ContentId> element MUST have no child elements.

2.2.2.3 ContentLocation

The <ContentLocation> element is an optional child element of the **Attachment** type that contains the relative URL for an attachment, and is used to match a reference to an inline attachment in an HTML message to the attachment in the attachments table.

The value of this element is a **string** ([MS-ASDTYPE] section 2.1) value.

A command response MUST have a maximum of one <ContentLocation> element per **Attachment** type. Command requests MUST NOT include the <ContentLocation> element. If a command request includes this element, the server returns a protocol status error.

The <ContentLocation> element MUST have no child elements.

2.2.2.4 Data

The <Data> element is an optional child element of the **Body** type and the **BodyPart** type that contains the data of the **message body** or the message part of the calendar item, contact, document, e-mail, or task.

The value of this element is a **string** ([MS-ASDTYPE] section 2.1)

A command response MUST have a maximum of one <Data> element within each returned **Body** and **BodyPart** container element. Command requests MAY include the <Data> element.

In a response, the <Data> element MUST have no child elements.

The content of the <Data> element is returned as a **string** in the format that is specified by the <Type> element. If the <Truncated> element is included in the response, then the data in the <Data> element is truncated. The <EstimatedDataSize> element provides a rough estimation of the actual size of the complete content of the <Data> **string**. The client makes appropriate buffer provisions to handle the incoming data. If the <AllOrNone> element of the **Search** command is included, and there is no **Body** or **BodyPart** type (also set by the **Search** command) that the server can fall back to, then the server responds with a protocol status error of 2.

2.2.2.5 DisplayName

The <DisplayName> element is an optional child element of the **Attachment** type that specifies the display name of the attachment.

The value of this element is a **string** value ([\[MS-ASDTYPE\]](#) section 2.1).

A command response MUST have a maximum of one <DisplayName> element per **Attachment** type. Command requests MUST NOT include the <DisplayName> element. If a command request includes this element, then the server returns a protocol status error.

The <DisplayName> element MUST have no child elements.

2.2.2.6 EstimatedDataSize

The <EstimatedDataSize> element is an optional child element of the **Body** type and a required child element of the **BodyPart** type and the **Attachment** type. For the **Body** type, this element provides an informational estimate of the size of the data associated with the **Body** type. For the **BodyPart** type, this element provides an informational estimate of the complete size of the unique message part of the e-mail. The <EstimatedDataSize> element SHOULD be included in a response message whenever the <Truncated> element is set to TRUE.

The value of this element is an **integer** value ([\[MS-ASDTYPE\]](#) section 2.2). If the client includes a noninteger value for this element in a command request, the server responds with a status error of 6.

A command response MUST have a maximum of one <EstimatedDataSize> element per **Body**, **BodyPart**, or **Attachment** type. If an <EstimatedDataSize> element is included in a request, then the element is ignored and no error is thrown.

The <EstimatedDataSize> element MUST have no child elements.

The <EstimatedDataSize> value represents the original size of the content in the **store** for the **Body** and **Attachment** types and the complete size of the unique message part content for the **BodyPart** type, and is specified in bytes. The client uses this number only for an informational display to the user. This number is only an estimate and the actual size of the body when fetched can differ based on the content filtering **rules** applied. The client does not make any memory allocations based on this number for future requests.

2.2.2.7 FileReference

The <FileReference> element specifies a unique **identifier** that is assigned by the server to each attachment to a message. In an **ItemOperations** command request ([\[MS-ASCMD\]](#) section 2.2.1.8), the <FileReference> element is an optional child element of the <Fetch> element ([\[MS-ASCMD\]](#) section 2.2.1.8.2.2). In a **Sync** command response, ([\[MS-ASCMD\]](#) section 2.2.1.19) the <FileReference> element is a required child element of the **Attachment** type (section [2.2.1.2](#)).

The value of this element is a **string** value ([\[MS-ASDTYPE\]](#) section 2.1). If the client includes a zero-length string for the value of this element in an **ItemOperations** command request, the server responds with a protocol status error of 15.

The <FileReference> element MUST have no child elements.

2.2.2.8 IsInline

The <IsInline> element is an optional child element of the **Attachment** type that specifies whether the attachment is embedded in the message.

The value of this element is a **boolean** value ([\[MS-ASDTYPE\]](#) section 2.3).

A command response MUST have a maximum of one <IsInline> element per **Attachment** type. Command requests MUST NOT include the <IsInline> element. If this element is included in a command request, then the server returns a protocol status error.

The <IsInline> element MUST have no child elements.

2.2.2.9 Method

The <Method> element is a required child element of the **Attachment** type that identifies the method in which the attachment was attached.

The value of this element is an **enumeration** value ([\[MS-ASDTYPE\]](#) section 2.9).

A command response MUST have a maximum of one <Method> element per **Attachment** type. Command requests MUST NOT include the <Method> element. If a command request includes this element, the server returns a protocol status error.

The <Method> element MUST have no child elements.

The following table defines the valid values of the <Method> enumeration.

Value	Meaning	Notes
1	Normal attachment	The attachment is a normal attachment. This is the most common value.
2	Reserved	Do not use.
3	Reserved	Do not use.
4	Reserved	Do not use.
5	Embedded message	Indicates that the attachment is an e-mail message, and that the attachment file has an .eml extension.
6	Attach OLE	Indicates that the attachment is an embedded OLE object, such as an inline image.

2.2.2.10 NativeBodyType

The <NativeBodyType> element is a required child element of the <ApplicationData> element ([\[MS-ASCMD\]](#) section 2.2.1.19.1.7) in the **Sync** command that specifies the original format type of the item.

The value of this element is an **enumeration** value ([\[MS-ASDTYPE\]](#) section 2.9). If the client includes a non-enumerated value type for this element, then the server responds with a protocol status error of 6.

A command response MUST have a maximum of one <NativeBodyType> element per <ApplicationData> element. Command requests MAY include the <NativeBodyType> element.

The <NativeBodyType> element MUST have no child elements. If a client request includes child elements with this element, then the server responds with a protocol status error.

The following table defines the valid values of the <NativeBodyType> enumeration.

Enumeration Value	Description
1	Plain text
2	HTML
3	RTF

The <NativeBodyType> and <Type> elements have the same value unless the server has modified the format of the body to match the client's request. The client can specify a preferred body format by using the <Type> element of a **Search** or **Sync** command request.

2.2.2.11 Preview

The <Preview> element [<2>](#) is an optional child element of the **BodyPreference** type, the **BodyPartPreference** type, the **Body** type, and the **BodyPart** type.

The <Preview> element of a **BodyPreference** type or a **BodyPartPreference** type in a command request specifies the maximum length of the **Unicode** plain text message or message part preview to be returned to the client. The value of this element is an **integer** value ([\[MS-ASDTYPE\]](#) section 2.2). If the client includes a non-**integer** value for this element, then the server responds with a protocol status error. This element **MUST** have a value set from 0 to 255, inclusive. If the value of this element exceeds 255, then the server responds with a protocol status error.

The <Preview> element of the **Body** type or the **BodyPart** type contains the Unicode plain text message or message part preview returned to the client. The value of this element is a **string** ([\[MS-ASDTYPE\]](#) section 2.1). The Preview element in a response **MUST** contain no more than the number of characters specified in the request. The Preview element **MUST** be returned in a command response if a **BodyPreference** type or a **BodyPartPreference** type in the request included a Preview element and the server can honor the request.

A command request **MUST** have a maximum of one <Preview> element per **BodyPreference** type and **BodyPartPreference** type. If a command request contains more than one <Preview> element per **BodyPreference** type or **BodyPartPreference** type, then the server returns a protocol status error of 2. Command responses **MUST** have a maximum of one <Preview> element per **Body** type and **BodyPart** type.

The <Preview> element **MUST** have no child elements. If a client request includes child elements in the <Preview> element, then the server responds with a protocol status error.

2.2.2.12 Status

The <Status> element indicates the success or failure of the response in returning <Data> element content in the **BodyPart** given the **BodyPartPreference** settings in the request.

The following table lists valid values for the <Status> element.

Value	Meaning
1	Success.
176	The message part is too large.

2.2.2.13 Truncated

The <Truncated> element is an optional child element of the **Body** type and the **BodyPart** type that specifies whether the body of the item has been truncated according to the **BodyPreference** or **BodyPartPreference**, respectively, indicated by the client.

The value of this element is a **boolean** value ([\[MS-ASDTYPE\]](#) section 2.3). If the client includes a value other than a **boolean** value for this element in a command request, the server responds with a protocol status error of 6.

If a <Truncated> element is included in a command request, then it is ignored by the server.

A command response MUST have a maximum of one <Truncated> element per **Body** type or **BodyPart** type. If the command response does not contain a <Truncated> element, then the client MUST process the message as if the whole body is contained in the message response (as if the <Truncated> is set to FALSE).

The <Truncated> element MUST have no child elements. If a command request includes child elements in the <Truncated> element, then the server responds with a protocol status error of 6.

2.2.2.14 TruncationSize

The <TruncationSize> element is an optional child element of the **BodyPreference** type and the **BodyPartPreference** type that specifies the size, in bytes, of the content that the user wants to **Search**, **Synchronize**, or **Fetch**.

The value of this element is an **integer** value ([\[MS-ASDTYPE\]](#) section 2.2). If a client request submits a noninteger value for this element, the server responds with a protocol status error.

A command request MUST have a maximum of one <TruncationSize> element per **BodyPreference** type and **BodyPartPreference** type. If the client includes more than one <TruncationSize> element per **BodyPreference** type or **BodyPartPreference** type, the server responds with a protocol status error of 2.

Command responses MUST NOT include the <TruncationSize> element.

The <TruncationSize> element MUST have no child elements. If the client includes child elements for the <TruncationSize> element in a command request, the server responds with a protocol status error.

If the <TruncationSize> element is absent, then the server assumes that the user will search the entire content. If the server sends the response in the corresponding content type and if the size of the entire content is greater than the value that is specified by <TruncationSize>, then the server truncates the message to the size that is specified by <TruncationSize> and sends it in the specified encoding. The maximum value for <TruncationSize> is 4,294,967,295. If a client request includes a larger value for this element in a command request, the server responds with a protocol status error of 6.

2.2.2.15 Type

The <Type> element indicates the format type of the body content of the item. In a request, the <Type> element is a required child element of the **BodyPreference** type and the **BodyPartPreference** type. [<3>](#) If the <Type> element is not included in a command request, the server responds with a protocol status error. In a response, the <Type> element is a required child element of the **Body** type and the **BodyPart** type.

The value of this element is an **enumeration** value ([MS-ASDTYPE] section 2.9). If the client submits a value other than an **enumeration** value for this element in a command request, the server responds with a protocol status error of 2.

A command request or response MUST have a maximum of one <Type> element per **BodyPreference**, **BodyPartPreference**, **Body**, or **BodyPart** type. If a client request includes more than one <Type> element per **BodyPreference**, **BodyPartPreference**, **Body**, or **BodyPart**, the server responds with a protocol status error of 2.

The <Type> element MUST have no child elements. If the client issues a request with child elements, the server responds with a protocol status error.

The following table defines the valid values of the <Type> enumeration. Only a value of 2 (HTML) SHOULD<4> be used in the <Type> element of a **BodyPartPreference** request. If the client includes a non-enumerated value for this element in a command request, the server responds with a protocol status error.

Enumeration Value	Description
1	Plain text
2	HTML
3	RTF
4	MIME

Additional <Type> values can be used if the client and the server support the content type. If the server or client receives a <Type> value that it does not support or understand, it skips the entire parent element without processing it.

2.2.3 Attributes

This specification does not define any common XML schema attribute definitions.

2.2.4 Groups

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

Attribute	Description
TopLevelSchemaProps	Identifies the Body , BodyPart , and Attachment types as being part of the TopLevelSchemaProps group.

2.2.4.1 TopLevelSchemaProps

The **TopLevelSchemaProps** group identifies the **Body**, **BodyPart**, and **Attachment** types as being part of the **TopLevelSchemaProps** group.

2.2.5 Attribute Groups

This specification does not define any common XML schema attribute group definitions.

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 specified in this document.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Higher-Layer Triggered Events

None.

3.1.5 Message Processing Events and Sequencing Rules

3.1.5.1 Commands

Command	Description
Fetch	Retrieves an item from the server.
Search	Retrieves entries from the store.
Sync	Synchronizes changes in a collections set between the client and the server.

3.1.5.1.1 Fetch

The request message for the **Fetch** command can include the following elements and types: **BodyPreference**, **BodyPartPreference**, <Type>, <TruncationSize>, <FileReference>, and <AllOrNone>.

3.1.5.1.2 Search

The request message for the **Search** command can include the following elements and types: **BodyPreference**, **BodyPartPreference**, <Type>, <TruncationSize>, and <AllOrNone>.

3.1.5.1.3 Sync

The request message for the **Sync** command can include the following elements and types:

- **BodyPreference** (section [2.2.1.4](#)) and **BodyPartPreference** (section [2.2.1.6](#)), along with the child elements <Type> (section [2.2.2.15](#)), <TruncationSize> (section [2.2.2.14](#)), <Preview> (section [2.2.2.11](#)), and <AllOrNone> (section [2.2.2.1](#))

- <Attachments> (section [2.2.1.1](#)), along with its child element <Attachment> (section [2.2.1.2](#)). <Attachment> can include the child elements <FileReference> (section [2.2.2.7](#)), <Method> (section [2.2.2.9](#)), and <EstimatedDataSize> (section [2.2.2.6](#)).

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 specified in this document.

The abstract data model used by the server and the client are the same.

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

3.2.5.1 Commands

Command	Description
Sync	Synchronizes changes in a collections set between the client and the server.

3.2.5.1.1 Sync

The response message for the **Sync** command can include the following: **Attachment**, <DisplayName>, <FileReference>, <Method>, <EstimatedDataSize>, <ContentId>, <ContentLocation>, <IsInline>, **Body**, **BodyPart**, <Type>, <Truncated>, <Data>, and <NativeBodyType>.

4 Protocol Examples

For examples of the **Search** command utilizing the AirSyncBase Namespace protocol, see [\[MS-ASCMD\]](#) sections [4.7.3.1](#) and [4.7.3.2](#). For examples of the **Fetch** command utilizing the AirSyncBase Namespace protocol, see [\[MS-ASCMD\]](#) sections [4.7.1.1](#) and [4.7.2](#).

PRELIMINARY

5 Security

5.1 Security Considerations for Implementers

None.

5.2 Index of Security Parameters

None.

PRELIMINARY

6 Appendix A: Product Behavior

The information in this specification is applicable to the following product versions. References to product versions include released service packs.

- Microsoft® Exchange Server 2007
- Microsoft® Exchange Server 2010
- Microsoft® Exchange Server 2010 SP1 Beta

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.

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 2.2.1.6:](#) Only a value of 2 (HTML) in the <Type> element of a **BodyPartPreference** request is supported by the server when the MS-ASProtocolVersion header is set to 14.1. The **BodyPartPreference** request is not supported when the MS-ASProtocolVersion header is set to 12.1 or 14.0.

[<2> Section 2.2.2.11:](#) The Preview element is not supported when the MS-ASProtocolVersion header is set to 12.1.

[<3> Section 2.2.2.15:](#) The **BodyPartPreference** type is not supported when the MS-ASProtocolVersion header is set to 12.1 or 14.0.

[<4> Section 2.2.2.15:](#) When the MS-ASProtocolVersion header is set to 14.1, the server only supports a <Type> value of 2 in a **BodyPartPreference** request.

7 Change Tracking

This section identifies changes made to [MS-ASAIRS] protocol documentation between February 2010 and May 2010 releases. Changes are classed as major, minor, or editorial.

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.
- A protocol is deprecated.
- The removal of a document from the documentation set.
- Changes made for template compliance.

Minor changes do not affect protocol interoperability or implementation. Examples are updates to fix technical accuracy or ambiguity at the sentence, paragraph, or table level.

Editorial changes apply to grammatical, formatting, and style issues.

No changes means that the document is identical to its last release.

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

- New content added.
- Content update.
- 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 always have the revision 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.

Changes are listed in the following table. If you need further information, please contact protocol@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Revision Type
1.1 Glossary	54679 Added "message part" to list of terms defined in [MS-OXGLOS].	Y	New content added due to protocol revision.
1.1 Glossary	Added "Unicode" to list of glossary terms defined in [MS-OXGLOS].	N	Content update.
1.3 Overview	Updated the section title.	N	Content updated for template compliance.
1.4 Relationship to Other Protocols	51564 Included description of this protocol specification's relationships to the ActiveSync content classes and to [MS-ASDTYPE].	Y	Content update.
2.2 Message Syntax	54187 Added the new BodyPartPreference type.	Y	New content added due to protocol revision.
2.2 Message Syntax	54679 Added the new BodyPart type.	Y	New content added due to protocol revision.
2.2 Message Syntax	Added the Preview element to Body and BodyPart complex types.	Y	Protocol syntax updated.
2.2 Message Syntax	54002 Added XSD for description of Data element as optional.	N	Content update.
2.2 Message Syntax	53969 Removed the Restriction element from XSD, as it is not used by any ActiveSync commands.	Y	Content removed.
2.2.1	54187	Y	New content

Section	Tracking number (if applicable) and description	Major change (Y or N)	Revision Type
Complex Types	Added description of new BodyPartPreference type.		added due to protocol revision.
2.2.1 Complex Types	54679 Added new BodyPart complex type.	Y	New content added due to protocol revision.
2.2.1 Complex Types	51579 Clarified the definition of the Body type.	N	Content update.
2.2.1.1 Attachments	53709 Clarified that the type is optional in a command request.	Y	Content update.
2.2.1.1 Attachments	54183 Specified the server response when the client includes this element in a command request.	N	Content update.
2.2.1.2 Attachment	54188 Specified the server response if the client includes this element in a command request.	N	Content update.
2.2.1.3 Body	48146 Changed description of Data element within Body type from required to optional.	N	Protocol syntax updated.
2.2.1.3 Body	54000 Updated Body to include Preview element.	Y	Content update.
2.2.1.3 Body	54002 Added description of Data element as optional.	N	Content update.
2.2.1.3 Body	53726 Modified reference to ApplicationData in [MS-ASCMD].	N	Content update.
2.2.1.3 Body	53751 Clarified that the Truncated element is allowed but non-functional in a command request.	N	Content update.
2.2.1.3 Body	51579 Clarified the definition of this element.	N	Content update.
2.2.1.3 Body	54230 Specified that the inclusion of this element signifies the existence of one or more variable-length data fields for the associated item.	N	Content update.
2.2.1.4 BodyPreference	54249 Specified the server response when this element is not a child of the Options element in a request.	N	Content update.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Revision Type
2.2.1.4 BodyPreference	54276 Specified the server response when the client issues a command request with the child elements of this type in the improper order.	N	Content update.
2.2.1.4 BodyPreference	54277 Specified the server response when a client request includes two or more BodyPreference types for the same value of the Type element.	N	Content update.
2.2.1.4 BodyPreference	51577 Clarified that the child elements of this element are specific to command requests, and not command responses.	N	Content update.
2.2.1.4 BodyPreference	54009 Clarified how the options set by this type are stored by the server and applied to subsequent requests.	Y	Content update.
2.2.1.5 BodyPart	54679 Added section for new BodyPart element.	Y	New content added due to protocol revision.
2.2.1.6 BodyPartPreference	54187 Added section for new BodyPartPreference type.	N	New content added due to protocol revision.
2.2.2 Elements	54187 Added reference to new BodyPartPreference type.	Y	New content added due to protocol revision.
2.2.2 Elements	55107 Added Status element of BodyPart complex type.	Y	New content added due to protocol revision.
2.2.2 Elements	51579 Clarified the definition of the EstimatedDataSize element.	N	Content update.
2.2.2.1 AllOrNone	54187 Added a reference to new the BodyPartPreference type.	Y	New content added due to protocol revision.
2.2.2.1 AllOrNone	54281 Specified the server response when the client includes more than one instance of this element in a request.	N	Content update.
2.2.2.1 AllOrNone	54279 Specified the server response when the client includes a non-Boolean value for this element in a command request.	N	Content update.
2.2.2.1	51583	Y	Content update.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Revision Type
AllOrNone	Clarified how the client uses this element to instruct the server that it does not want partial data.		
2.2.2.2 ContentId	54283 Specified the server response if the client includes this element in a command request.	N	Content update.
2.2.2.3 ContentLocation	54295 Clarified the server response if this element is included in a command request.	N	Content update.
2.2.2.4 Data	54679 Added a reference to the new BodyPart type.	Y	New content added due to protocol revision.
2.2.2.4 Data	54002 Clarified that the Data element is optional.	N	Content update.
2.2.2.4 Data	53967 Clarified the server response when a client request includes the AllOrNone element without a Body or BodyPart type.	N	Content update.
2.2.2.5 DisplayName	54315 Clarified the server response when this element is included in a command request.	N	Content update.
2.2.2.6 EstimatedDataSize	54679 Added reference to new BodyPart type.	Y	New content added due to protocol revision.
2.2.2.6 EstimatedDataSize	54195 Specified the server response if the client includes a noninteger value for this element in a command request.	N	Content update.
2.2.2.6 EstimatedDataSize	53954 Removed statements declaring that this element is only included when the Data element is truncated.	Y	Content removed.
2.2.2.6 EstimatedDataSize	51579 Clarified the definition of this element.	N	Content update.
2.2.2.7 FileReference	54189 Specified the server response if the client includes a zero-length string as the value of this element in an ItemOperations command request.	N	Content update.
2.2.2.8 IsInline	54312 Clarified the server response when this element is included in a command request.	N	Content update.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Revision Type
2.2.2.9 Method	54310 Clarified the server response if a command request includes this element.	N	Content update.
2.2.2.10 NativeBodyType	54298 Specified the server response when the client includes child elements of this element in a command request.	N	Content update.
2.2.2.10 NativeBodyType	54309 Specified the server response when the client includes a non-enumerated value type for this element.	N	Content update.
2.2.2.11 Preview	54679 Added a reference to the new BodyPart type.	Y	New content added due to protocol revision.
2.2.2.11 Preview	54187 Added a reference to the new BodyPartPreference type.	Y	New content added due to protocol revision.
2.2.2.11 Preview	54000 Added content from previous section 2.2.2.15 and removed the section.	Y	Content update.
2.2.2.11 Preview	54001 Updated references to possible containing elements for this element.	N	Content update.
2.2.2.11 Preview	Clarified the conditions under which this element is not supported.	N	Content update.
2.2.2.11 Preview	54282 Specified the server response when the client includes child elements within this element.	N	Content update.
2.2.2.11 Preview	54327 Specified the server response when a client request contains more than one Preview element per allowed parent type.	N	Content update.
2.2.2.11 Preview	54284 Specified the server response when the client includes a value for this element in excess of 255.	N	Content update.
2.2.2.11 Preview	54285 Specified the server response when the client includes a non-integer value for this element.	N	Content update.
2.2.2.12 Status	55107 Added section for Status element of new BodyPart complex type.	Y	Content updated due to protocol revision.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Revision Type
2.2.2.13 Truncated	54187 Added a reference to the new BodyPartPreference type.	Y	New content added due to protocol revision.
2.2.2.13 Truncated	54679 Added a reference to the new BodyPart type.	Y	New content added due to protocol revision.
2.2.2.13 Truncated	54331 Removed a sentence stating that this element is not allowed in a command request.	N	Content removed.
2.2.2.13 Truncated	54271 Specified the server response when a client request includes child elements within this element.	N	Content update.
2.2.2.13 Truncated	54232 Specified that this element is ignored by the server in a command request.	N	Content update.
2.2.2.13 Truncated	54280 Specified the server response when the client submits a non-Boolean value for this element in a command request.	N	Content update.
2.2.2.14 TruncationSize	54187 Added reference to new BodyPartPreference type.	Y	New content added due to protocol revision.
2.2.2.14 TruncationSize	54269 Specified how the server responds if the client submits a noninteger value for this element.	N	Content update.
2.2.2.14 TruncationSize	54257 Specified the server response if a client includes child elements for this element in a command request.	N	Content update.
2.2.2.14 TruncationSize	54255 Specified the server response if a client request exceeds the maximum value for this element.	N	Content update.
2.2.2.14 TruncationSize	54260 Specified the server response if the client includes more than one instance of this element per supported parent type.	N	Content update.
2.2.2.15 Type	54679 Added reference to new BodyPart type.	Y	New content added due to protocol revision.
2.2.2.15 Type	54187 Added reference to new BodyPartPreference	Y	New content added due to

Section	Tracking number (if applicable) and description	Major change (Y or N)	Revision Type
	type.		protocol revision.
2.2.2.15 Type	54237 Specified the server response if a client request includes child elements in this element.	N	Content update.
2.2.2.15 Type	54248 Specified the server response if a client request does not include this element.	N	Content update.
2.2.2.15 Type	54247 Specified the server response if a client request includes a non-enumerated value.	Y	Content update.
2.2.2.15 Type	54244 Specified the server response if a client request includes more than one of this element per supported parent type.	N	Content update.
2.2.2.15 Type	55263 Specified restriction on Type in BodyPartPreference for server behavior.	N	Content updated due to protocol revision.
2.2.4 Groups	54679 Added reference to new BodyPart type.	Y	New content added due to protocol revision.
2.2.4 Groups	54007 Updated child elements of TopLevelSchemaProps with current XSD.	N	Content update.
2.2.4.1 TopLevelSchemaProps	54679 Added reference to new BodyPart type.	Y	New content added due to protocol revision.
3.1.5.1.1 Fetch	54187 Added reference to new BodyPartPreference type.	Y	New content added due to protocol revision.
3.1.5.1.2 Search	54187 Added a reference to the new BodyPartPreference type.	Y	New content added due to protocol revision.
3.1.5.1.2 Search	54008 Removed the FileReference element from the list of allowable child elements.	N	Content removed.
3.1.5.1.3 Sync	54187 Added a reference to the new BodyPartPreference type.	Y	New content added due to protocol revision.
3.1.5.1.3 Sync	54006 Clarified which elements are supported in a command request.	Y	Content update.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Revision Type
3.2.5.1.1 Sync	54679 Added reference to new BodyPart type.	Y	New content added due to protocol revision.

PRELIMINARY

8 Index

A

Abstract data model
[client](#) 21
[server](#) 22
[Applicability](#) 6

C

[Change tracking](#) 26
Client
[abstract data model](#) 21

D

Data model – abstract
[client](#) 21
[server](#) 22

E

[Examples - overview](#) 23

F

[Fields - vendor-extensible](#) 6

G

[Glossary](#) 5

I

[Introduction](#) 5

M

Messages
[overview](#) 8
[syntax](#) 8
[transport](#) 8

N

[Normative references](#) 5

O

[Overview](#) 6

P

[Preconditions](#) 6
[Prerequisites](#) 6
[Product behavior](#) 25

R

References

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

S

Security
[overview](#) 24
Server
[abstract data model](#) 22
Syntax
[messages - overview](#) 8

T

[Tracking changes](#) 26
[Transport](#) 8

V

[Vendor-extensible fields](#) 6