

[MS-ASDOC]: ActiveSync Document Class Protocol Specification

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft's Open Specification Promise (available here: <http://www.microsoft.com/interop/osp>) or the Community Promise (available here: <http://www.microsoft.com/interop/cp/default.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 iplq@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
12/03/2008	1.0.0	Major	Initial Release.
02/04/2009	1.0.1	Editorial	Revised and edited technical content.
03/04/2009	1.0.2	Editorial	Revised and edited technical content.
04/10/2009	2.0.0	Major	Updated applicable product releases.
07/15/2009	3.0.0	Major	Revised and edited for technical content.
11/04/2009	4.0.0	Major	Updated and revised the technical content.
02/10/2010	5.0.0	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 Protocol Overview	6
1.4 Relationship to Other Protocols.....	6
1.5 Prerequisites/Preconditions.....	6
1.6 Applicability Statement.....	6
1.7 Versioning and Capability Negotiation.....	6
1.8 Vendor-Extensible Fields	6
1.9 Standards Assignments	6
2 Messages	7
2.1 Transport.....	7
2.2 Message Syntax.....	7
2.2.1 Complex Types.....	7
2.2.2 Elements.....	7
2.2.2.1 LinkId	8
2.2.2.2 DisplayName.....	8
2.2.2.3 IsFolder	8
2.2.2.4 CreationDate.....	8
2.2.2.5 LastModifiedDate.....	9
2.2.2.6 IsHidden	9
2.2.2.7 ContentLength	9
2.2.2.8 ContentType	9
3 Protocol Details.....	10
3.1 Client Details.....	10
3.1.1 Abstract Data Model.....	10
3.1.2 Timers	10
3.1.3 Initialization	10
3.1.4 Higher-Layer Triggered Events	10
3.1.4.1 Searching for Documents	10
3.1.4.2 Requesting Details for a Specific Document	10
3.1.4.3 Requesting the Document Body from the Server.....	10
3.1.5 Message Processing Events and Sequencing Rules	10
3.1.5.1 ItemOperations Command Request	11
3.1.5.2 Search Command Request.....	11
3.1.6 Timer Events.....	11
3.1.7 Other Local Events	11
3.2 Server Details.....	11
3.2.1 Abstract Data Model.....	11
3.2.2 Timers	12
3.2.3 Initialization	12
3.2.4 Higher-Layered Triggered Events.....	12
3.2.4.1 Searching for Documents	12
3.2.4.2 Requesting Details for a Specific Document	12
3.2.4.3 Requesting the Document Body from the Server.....	12
3.2.5 Message Processing Events and Sequencing Rules	12

3.2.5.1	ItemOperations Command Response	12
3.2.5.2	Search Command Response	13
3.2.6	Timer Events.....	13
3.2.7	Other Local Events	13
4	Protocol Examples	14
4.1	Searching for a Document by LinkID.....	14
4.2	Retrieving The Text of Document Using ItemOperations Command	15
4.3	Browsing a Document Folder	16
5	Security.....	18
5.1	Security Considerations for Implementers.....	18
5.2	Index of Security Parameters	18
6	Appendix A: Product Behavior	19
7	Change Tracking	20
8	Index.....	22

1 Introduction

ActiveSync supports accessing documents stored in Windows Sharepoint Services and on file shares specified using Universal Naming Convention (UNC) paths. The Document **class** protocol specifies how such document data is communicated from the server to the client in the ActiveSync protocol.

1.1 Glossary

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

class
collection
Coordinated Universal Time (UTC)
folder
Multipurpose Internet Mail Extensions (MIME)
Uniform Resource Identifier (URI)
WAP Binary XML (WBXML)
XML
XML schema

The following terms are specific to this document:

Uniform Naming Convention (UNC): A format for referencing an object, such as a folder or computer, that is shared across a local area network (LAN).

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

1.2 References

1.2.1 Normative References

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

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

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

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

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

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

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

[RFC822] Crocker, D.H., "STANDARD FOR THE FORMAT OF ARPA INTERNET TEXT MESSAGES", RFC 822, August 1982, <http://www.ietf.org/rfc/rfc0822.txt>.

[XML] Bray, T., et al., "Extensible Markup Language (XML) 1.0 (Fifth Edition)", November 2008, <http://www.w3.org/TR/REC-xml/>.

1.2.2 Informative References

None.

1.3 Protocol Overview

The Document class protocol specifies the **X**ML representation of documents used for client and server communication as specified in [\[MS-ASCMD\]](#).

1.4 Relationship to Other Protocols

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

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

1.5 Prerequisites/Preconditions

None.

1.6 Applicability Statement

This protocol specifies a set of elements and complex types for use in communicating document data using the commands specified in [\[MS-ASCMD\]](#). This set of elements and complex types is applicable when communicating document data such as the document's name, location, estimated size, and visibility between a mobile device and a server. These elements and complex types are not applicable when sending calendar, e-mail, note, contact, or task data between a mobile device and a server.

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

The Document class consists of a series of XML elements that are embedded inside of a command or a **collection** sent in accordance with [\[MS-ASCMD\]](#). The XML block containing the class elements is transmitted in either the request body of a request, or the response body of a response.

The types and elements of the Document class are defined in two namespaces: Doc and AirSyncBase. All of the Document class types and elements are specified in this document; however, complex types and elements defined in the AirSyncBase namespace are further specified in [\[MS-ASAIRS\]](#).

The parent element of the Document class elements depends upon the ActiveSync protocol command used to retrieve the class data. Commands and parent elements for the Document class **XML schema** are specified in section [3.1.5](#).

2.2 Message Syntax

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

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

The XML schema definition for the Document class in ActiveSync is as follows.

```
<?xml version="1.0" ?><xs:schema xmlns:tns="DOC:" attributeFormDefault="unqualified"
elementFormDefault="qualified"
targetNamespace="DOC:" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:A="AirSyncBase:">
  <xs:import namespace="AirSyncBase" />
  <xs:element name="LinkId" type="xs:string" />
  <xs:element name="DisplayName" type="xs:string" />
  <xs:element name="IsFolder" type="xs:unsignedByte" />
  <xs:element name="CreationDate" type="xs:dateTime" />
  <xs:element name="LastModifiedDate" type="xs:dateTime" />
  <xs:element name="IsHidden" type="xs:unsignedByte" />
  <xs:element name="ContentLength" type="xs:unsignedByte" />
  <xs:element name="ContentType" type="xs:unsignedByte" />
</xs:schema>
```

2.2.1 Complex Types

There are no complex types defined for the **document** class.

2.2.2 Elements

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

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

Value	Description
LinkId	The link to the document, specified as a Uniform Resource Identifier (URL) .
DisplayName	The name of the document, as displayed by the client.
IsFolder	Specifies whether the item is a folder or a document.
CreationDate	The date and time when the document was first created.
LastModifiedDate	The date and time when the document or its properties was last modified.
IsHidden	Specifies whether this is a hidden object.
ContentLength	The estimated size of the document, in bytes.
ContentType	The Multipurpose Internet Mail Extension (MIME) type of the binary- or base-64-encoded content.

2.2.2.1 LinkId

The <LinkId> element is an optional element that specifies the link to the document in the form of a **URI**.

2.2.2.2 DisplayName

The <DisplayName> element is an element that specifies the name of the document as it is displayed to the user.

The <DisplayName> element is not included in a command request.

A **Search** command response has a minimum of one <DisplayName> element per response. For more information on the **Search** command, see section [3.2.5.2](#).

2.2.2.3 IsFolder

The <IsFolder> element is an element that specifies whether the item is a folder.

The <IsFolder> element is not required in a command request.

The <IsFolder> element is required in a **Search** command response. For more information on the **Search** command, see section [3.2.5.2](#).

Valid values for this element are as follows.

Value	Description
0	The item is not a folder.
1	The item is a folder.

2.2.2.4 CreationDate

The <CreationDate> element is an element that specifies the date and time when the document was first created.

The <CreationDate> element is not included in a server request.

The <CreationDate> element is required in a server response.

The value of this element is in **Coordinated Universal Time (UTC)** format, as specified in [\[MS-ASDTYPE\]](#) section 2.6.

2.2.2.5 LastModifiedDate

The <LastModifiedDate> element is an element that specifies the date and time that the document or its properties was last modified.

The <LastModifiedDate> element is not included in a command request.

The <LastModifiedDate> element is required in a **Search** command response. For more information, see section [3.2.5.2](#).

The value of this element is in UTC format, as specified in [\[MS-ASDTYPE\]](#) section 2.6.

2.2.2.6 IsHidden

The <IsHidden> element is an element that specifies that the document or folder is a hidden object.

The <IsHidden> element **MUST NOT** be included in a command request. When it is included, the server **MUST** return a protocol error.

The <IsHidden> element is required in a **Search** command response.

The <IsHidden> element is an unsigned byte, as specified in [\[MS-ASDTYPE\]](#) section 2.10.

The value of the <IsHidden> element **MUST** be one of the following.

Value	Description
0	Is not hidden.
1	Is hidden.

2.2.2.7 ContentLength

The <ContentLength> element is an element that specifies the estimated size, in bytes, of the document.

The <ContentLength> element is not included in a command request.

The <ContentLength> element is required in a command response.

Since documents accessed using the ActiveSync protocol can be shared across a network, the value of the <ContentLength> element may differ between the time the document description is retrieved and the time the document itself is accessed.

2.2.2.8 ContentType

The <ContentType> element is an optional element that specifies the MIME type of the binary- or base64-encoded document, if known.

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.

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

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

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Higher-Layer Triggered Events

3.1.4.1 Searching for Documents

A client searches for Document class data on a server by sending a **Search** command request.

3.1.4.2 Requesting Details for a Specific Document

Document class data for one or more individual documents is requested by the client sending an **ItemOperations** command request, which is a wrapper for the **Fetch** command. An **ItemOperations** command can contain multiple **Fetch** commands.

3.1.4.3 Requesting the Document Body from the Server

The body of the document is not returned in the Document class. A client submits the value of the <LinkID> element (section [2.2.2.1](#)) in a separate **ItemOperations** request to obtain the body of the document as either base64-encoded text in the <Data> element of the response or as binary data, depending on the content type requested. Content type requests for the **ItemOperations** command are specified in section [3.1.5.1](#). The <Data> element is specified in [\[MS-ASCMD\]](#) section 2.2.1.8.3.13.

3.1.5 Message Processing Events and Sequencing Rules

The following sections define how various elements of the document class are used in the context of specific commands. For more details about the commands themselves, see [\[MS-ASCMD\]](#).

3.1.5.1 ItemOperations Command Request

A client uses the <ItemOperations> command to retrieve specific documents items from the server using the <Fetch> command. An **ItemOperations** request can contain multiple <Fetch> command.

Only the <LinkId> element (section [2.2.2.1](#)) can be included in an **ItemOperations** command request. The <LinkId> element (section [2.2.2.1](#)) is transmitted as a child of the <Fetch> element ([\[MS-ASCMD\]](#) section 2.2.1.8.2.2).

A client can use the HTTP header **MS-ASAcceptMultiPart: T** to specify that the server return the document data in binary format. Otherwise, the document is returned as text. This header is specified in [\[MS-ASCMD\]](#) section 2.2.1.8.1.

The **ItemOperations** command is specified in [\[MS-ASCMD\]](#) section 2.2.1.8.

3.1.5.2 Search Command Request

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

The <LinkId> element (section [2.2.2.1](#)) is required in a document search request. The <LinkId> element (section [2.2.2.1](#)) is transmitted as a child of the **EqualTo** type ([\[MS-ASCMD\]](#) section 2.2.1.14.1.13). The <Value> element ([\[MS-ASCMD\]](#) section 2.2.1.14.1.16) can also be included as a child of the **EqualTo** type ([\[MS-ASCMD\]](#) section 2.2.1.14.1.13). The value of the <Value> element is a string describing the **uniform naming convention (UNC)** path of a file on a file share. A full example of this usage is provided in [\[MS-ASCMD\]](#) section 4.10.1.

Search is specified in [\[MS-ASCMD\]](#) section 2.2.1.14.

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.

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

Command response: A WBXML formatted **Message** that adheres to the command schemas specified in [\[MS-ASCMD\]](#). The server **MUST** return a document class XML block for every e-mail that matches the criteria specified in the client command request. The server can return zero or more document class blocks in its response, depending on how many document items match the criteria specified in the client command request.

3.2.2 Timers

None.

3.2.3 Initialization

None.

3.2.4 Higher-Layered Triggered Events

3.2.4.1 Searching for Documents

A client searches for document class data on a server by sending a **Search** command request. The server responds with a **Search** command response.

3.2.4.2 Requesting Details for a Specific Document

Document class data for one or more individual documents is requested by the client sending an **ItemOperations** command request, which is a wrapper for the **Fetch** command. An **ItemOperations** command can contain multiple **Fetch** commands. The server responds with an **ItemOperations** command response.

3.2.4.3 Requesting the Document Body from the Server

The body of the document is not returned in the document class. A client can submit the value of the [LinkID](#) element in a separate **ItemOperations** request to obtain the body of the document. The server responds with an **ItemOperations** command response, which returns the body of the document as either base64-encoded text in the <Data> element of the response, or as binary text if the command request was a multi-part request. The <Data> element is specified in [\[MS-ASCMD\]](#) section 2.2.1.8.3.13.

3.2.5 Message Processing Events and Sequencing Rules

The following sections define how various elements of the document class are used in the context of specific commands. For more details about the commands themselves, see [\[MS-ASCMD\]](#).

3.2.5.1 ItemOperations Command Response

A client uses the **ItemOperations** command to retrieve specific document items from the server using the <Fetch> element. An **ItemOperations** request can contain multiple <Fetch> elements.

Any of the elements for the document class can be included in an **ItemOperations** command response.

Document class elements are returned either as children of the <Fetch> element, or as children of the <Properties> type ([\[MS-ASCMD\]](#) section 2.2.1.8.3.10).

If an **ItemOperations** command request for the body of the document was made using the **MS-ASAcceptMultiPart: T** header, the server MUST respond by providing the document body as binary data in multiple parts. Otherwise, the server MUST transmit the document as base64-encoded data within the [Data](#) type. The behavior of content delivery for documents is specified in [\[MS-ASCMD\]](#) section 2.2.1.8.1.

ItemOperations is specified in [\[MS-ASCMD\]](#) section 2.2.1.8.

3.2.5.2 Search Command Response

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

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

Document class elements are returned as children of the **Properties** type ([\[MS-ASCMD\]](#) section 2.2.1.14.2.2).

Search is specified in [\[MS-ASCMD\]](#) section 2.2.1.14.

3.2.6 Timer Events

None.

3.2.7 Other Local Events

None.

4 Protocol Examples

4.1 Searching for a Document by LinkID

The following example demonstrates searching for a document by matching its <LinkId> element (section [2.2.2.1](#)), which in this example is the **Uniform Naming Convention (UNC)** path of the document.

Request:

```
POST /Microsoft-Server-ActiveSync?Cmd=Search&User=deviceuser1&DeviceId=Device1&DeviceType=SmartPhone HTTP/1.1
Content-Type: application/vnd.ms-sync.wbxml
MS-ASProtocolVersion: 14.0
User-Agent: ASOM
Host: mail.contoso.com
Content-Length: 92

<?xml version="1.0" encoding="utf-8"?>
<Search xmlns="Search:" xmlns:A="DocumentLibrary:" >
  <Store>
    <Name>DocumentLibrary</Name>
    <Query>
      <EqualTo>
        <A:LinkId/>
        <Value>\\EXCH-D-810\DocumentShare\document.txt</Value>
      </EqualTo>
    </Query>
    <Options>
      <Range>0-999</Range>
    </Options>
  </Store>
</Search>
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ms-sync.wbxml
Date: Wed, 11 Nov 2009 18:07:38 GMT
Content-Length: 187

<?xml version="1.0" encoding="utf-8"?>
<Search xmlns:A0="AirSync:" xmlns:A1="POOMCONTACTS:" xmlns:A2="POOMMAIL:"
xmlns:A3="AirNotify:" xmlns:A4="POOMCAL:"
xmlns:A5="Move:" xmlns:A6="GetItemEstimate:" xmlns:A7="FolderHierarchy:"
xmlns:A8="MeetingResponse:"
xmlns:A9="POOMTASKS:" xmlns:A10="ResolveRecipients:" xmlns:A11="ValidateCert:"
xmlns:A12="POOMCONTACTS2:"
xmlns:A13="Ping:" xmlns:A14="Provision:" xmlns:A16="Gal:" xmlns:A17="AirSyncBase:"
xmlns:A18="Settings:"
xmlns:A19="DocumentLibrary:" xmlns:A20="ItemOperations:" xmlns:A21="ComposeMail:"
xmlns:A22="POOMMAIL2:"
xmlns:A23="Notes:" xmlns="Search:">
  <Status>1 Success</Status>
  <Response>
    <Store>
      <Status>1 Success</Status>
```

```

<Result>
  <Properties>
    <A19:LinkId>\\exch-d-810\DocumentShare\document.txt</A19:LinkId>
    <A19:DisplayName>document.txt</A19:DisplayName>
    <A19:IsFolder>0</A19:IsFolder>
    <A19:CreationDate>2009-11-11T17:07:08.156Z</A19:CreationDate>
    <A19:LastModifiedDate>2009-11-11T17:07:17.613Z</A19:LastModifiedDate>
    <A19:IsHidden>0</A19:IsHidden>
    <A19:ContentLength>13</A19:ContentLength>
    <A19:ContentType>text/plain</A19:ContentType>
  </Properties>
</Result>
<Range>0-0</Range>
<Total>1</Total>
</Store>
</Response>
</Search>

```

4.2 Retrieving The Text of Document Using ItemOperations Command

The following example demonstrates the client requesting the data for a Microsoft Word document using the **ItemOperations** command ([\[MS-ASCMD\]](#) section 2.2.1.8). (Note that, in the XML Response below, the value of the <Data> element has been truncated for the sake of brevity.)

Request:

```

POST /Microsoft-Server-ActiveSync?Cmd=ItemOperations&User=deviceuser1&DeviceId=Device1&DeviceType=Smart Phone
HTTP/1.1
Content-Type: application/vnd.ms-sync.wbxml
MS-ASProtocolVersion: 14.0
User-Agent: ASOM
Host: mail.contoso.com
Content-Length: 80

<?xml version="1.0" encoding="utf-8"?>
<ItemOperations xmlns:documentlibrary="DocumentLibrary:" xmlns="ItemOperations:" >
  <Fetch>
    <Store>DocumentLibrary</Store>
    <documentlibrary:LinkId>\\EXCH-D-810\DocumentShare\Word
Document.docx</documentlibrary:LinkId>
  </Fetch>
</ItemOperations>

```

Response:

```

HTTP/1.1 200 OK
Content-Type: application/vnd.ms-sync.wbxml
Date: Wed, 11 Nov 2009 19:16:21 GMT
Content-Length: 13373

<?xml version="1.0" encoding="utf-8"?>
<ItemOperations xmlns:A0="AirSync:" xmlns:A1="POOMCONTACTS:" xmlns:A2="POOMMAIL:"
xmlns:A3="AirNotify:"
xmlns:A4="POOMCAL:" xmlns:A5="Move:" xmlns:A6="GetItemEstimate:" xmlns:A7="FolderHierarchy:"
xmlns:A8="MeetingResponse:"

```

```

xmlns:A9="POOMTASKS:" xmlns:A10="ResolveRecipients:" xmlns:A11="ValidateCert:"
xmlns:A12="POOMCONTACTS2:"
xmlns:A13="Ping:" xmlns:A14="Provision:" xmlns:A15="Search:" xmlns:A16="Gal:"
xmlns:A17="AirSyncBase:"
xmlns:A18="Settings:" xmlns:A19="DocumentLibrary:" xmlns:A21="ComposeMail:"
xmlns:A22="POOMMAIL2:" xmlns:A23="Notes:"
xmlns="ItemOperations:">
  <Status>1</Status>
  <Response>
    <Fetch>
      <Status>1</Status>
      <A19:LinkId>\\EXCH-D-810\DocumentShare\Word Document.docx</A19:LinkId>
      <Properties>

<Data>UESDBBQABgIAAAAIQDd/+ImYzsKNchci+VLqQHEkJU4+RzBv1jKu6vsf0VwOamabaWQ1pZ+9AtcdYN1/WD103G
n4KZu/Yy4kVyAdhb9kuYipsSc
ZyjWop9SwabDDPJZ2RYqwKNuBpotX1RP9fi46FLAmhCYnP83x1nANaXg902aJ5x687HyFZLBZ9e/tDg7MvaD4BAAD//wM
AUESDBBQABgIAAAAIQDWZLNR
+gAAADEDEAAAcaAgBd29yZC9fcmVs cy9kb2N1bWVu dC54bWwu cmVscyCiBAEooAABAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAA...</Data>
      <Version>2009-11-11T19:15:45.177Z</Version>
    </Fetch>
  </Response>
</ItemOperations>

```

4.3 Browsing a Document Folder

The following example demonstrates browsing a folder on a remote share. The client submits a request for a folder to view, and the server responds with a list of the folder's contents.

Request:

```

POST /Microsoft-Server-ActiveSync?Cmd=Search&User=deviceuser1&DeviceId=Device1&DeviceType=SmartPhone HTTP/1.1
Content-Type: application/vnd.ms-sync.wbxml
MS-ASProtocolVersion: 14.0
User-Agent: ASOM
Host: mail.contoso.com
Content-Length: 316

<?xml version="1.0" encoding="utf-8"?>
<Search xmlns="Search:" xmlns:A="DocumentLibrary:">
  <Store>
    <Name>DocumentLibrary</Name>
    <Query>
      <EqualTo>
        <A:LinkId/>
        <Value>\\myserver\myshare</Value>
      </EqualTo>
    </Query>
    <Options>
      <Range>0-999</Range>
    </Options>
  </Store>
</Search>

```


Response:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.ms-sync.wbxml
Date: Wed, 11 Nov 2009 18:07:38 GMT
Content-Length: 1383

<?xml version="1.0" encoding="utf-8"?>
<Search xmlns="Search:" xmlns:A="DocumentLibrary:">
  <Status>1</Status>
  <Response>
    <Store>
      <Status>1</Status>
      <Result>
        <Properties>
          <A:LinkId>\myserver\myshare</A:LinkId>
          <A:DisplayName>d$</A:DisplayName>
          <A:IsFolder>1</A:IsFolder>
          <A:CreationDate>2007-10-02T00:34:28.686Z</A:CreationDate>
          <A:LastModifiedDate>2009-11-13T21:48:20.919Z</A:LastModifiedDate>
          <A:IsHidden>1</A:IsHidden>
        </Properties>
      </Result>
      <Result>
        <Properties>
          <A:LinkId>\myserver\myshare\blah.txt</A:LinkId>
          <A:DisplayName>blah.txt</A:DisplayName>
          <A:IsFolder>0</A:IsFolder>
          <A:CreationDate>2007-10-02T18:26:52.265Z</A:CreationDate>
          <A:LastModifiedDate>2009-04-02T02:57:55.843Z</A:LastModifiedDate>
          <A:IsHidden>1</A:IsHidden>
        </Properties>
      </Result>
      <Result>
        <Properties>
          <A:LinkId>\myserver\myshare\foo</A:LinkId>
          <A:DisplayName>foo</A:DisplayName>
          <A:IsFolder>1</A:IsFolder>
          <A:CreationDate>2009-10-13T00:43:44.660Z</A:CreationDate>
          <A:LastModifiedDate>2009-10-13T00:46:17.421Z</A:LastModifiedDate>
          <A:IsHidden>0</A:IsHidden>
        </Properties>
      </Result>
      <Range>0-2</Range>
      <Total>3</Total>
    </Store>
  </Response>
</Search>
```

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 product versions. References to product versions include released service packs.

- Microsoft Exchange Server 2007
- Microsoft Exchange Server 2010

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

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

7 Change Tracking

This section identifies changes made to [MS-ASDOC] protocol documentation between November 2009 and February 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	53677 Added Uniform Naming Convention (UNC) to the local glossary list.	Y	New content added.
2.2 Message Syntax	51394 Changed the type of the IsFolder element.	Y	Content update.
2.2.2.7 ContentLength	53806 Modified the element names for consistency.	N	Content update.
4 Protocol Examples	53677 Removed text.	N	Content update.
4.1 Searching for a Document by LinkID	53677 New topic.	Y	New content added.
4.2 Retrieving The Text of Document Using ItemOperations Command	53677 New topic.	Y	New content added.
4.3 Browsing a Document Folder	53677 New topic.	Y	New content added.

8 Index

A

[Applicability](#) 6

C

[Capability negotiation](#) 6

[Change tracking](#) 20

Client

[overview](#) 10

E

Examples

[overview](#) 14

G

[Glossary](#) 5

I

[implementer – security considerations](#) 18

[Index of security parameters](#) 18

[Informative references](#) 6

[Introduction](#) 5

M

Messages

[overview](#) 7

[transport](#) 7

N

[Normative references](#) 5

O

[Overview](#) 6

P

[Parameters – security index](#) 18

[Preconditions](#) 6

[Prerequisites](#) 6

[Product behavior](#) 19

R

References

[informative](#) 6

[normative](#) 5

[Relationship to other protocols](#) 6

S

Security

[implementer considerations](#) 18

[overview](#) 18

[parameter index](#) 18

Server

[overview](#) 11

[Standards assignments](#) 6

T

[Tracking changes](#) 20

[Transport](#) 7

V

[Vendor-extensible fields](#) 6

[Versioning](#) 6