[MS-ASCNTC]: ActiveSync Contact Class Protocol Specification

Revision Summary			
Author	Date	Version	Comments
Microsoft Corporation	December 3, 2008	1.0	Initial Release.
Microsoft Corporation	February 4, 2009	1.01	Revised and edited technical content.

Table of Contents

1 Introduction

Mobile devices that communicate by using the ActiveSync protocol need to exchange contact data. The ActiveSync Contact Class protocol specifies the ActiveSync protocol format for the interchange of contact data.

1.1 Glossary

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

class
collection
message database (MDB)
synchronization
Uniform Resource Identifier (URI)
WAP Binary XML (WBXML)
XML

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

[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 Protocol Specification", 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.

[RFC822] Crocker, D.H., "Standard for ARPA Internet Text Messages", RFC 822, August 1982, http://www.ietf.org/rfc/rfc0822.txt.[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, http://www.ietf.org/rfc/rfc2119.txt.

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

1.2.2 Informative References

None.

1.3 Protocol Overview

The ActiveSync Contact Class protocol specifies an **XML** representation of contact data as used in various ActiveSync commands.

1.4 Relationship to Other Protocols

The ActiveSync Contact Class protocol specifies an **XML** representation of contact data that is used by commands specified in [MS-ASCMD].

All data types in this document conform to the data type definitions specified in [MS-ASDTYPE]. Common XML schema elements used by other classes are defined in [MS-ASAIRS].

1.5 Prerequisites/Preconditions

None.

1.6 Applicability Statement

None.

1.7 Versioning and Localization

None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

The Contact **class** consists of a series of **XML** elements that are embedded inside of a **collection** transmitted according to the syntax of the ActiveSync protocol command used. The XML block containing the class elements is transmitted in either the request body of a request, or the response body of a response.

The parent element of the Contact class elements depends upon the ActiveSync protocol command used to retrieve class data. Commands and parent elements for the Contact class XML schema are specified in section 2.2.8.

This section describes the child elements that can be returned by an ActiveSync command. This section also describes concepts that are related to the Contact class.

2.2 Message Syntax

The markup MUST be well-formed XML, as specified in [XML].

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

The XML schema definition for the Contact class in ActiveSync is as follows:

```
<?xml version="1.0" ?>
<xs:schema xmlns:tns="CONTACTS:" attributeFormDefault="unqualified"
elementFormDefault="qualified" targetNamespace="CONTACTS:"
xmlns:xs=http://www.w3.org/2001/XMLSchema xmlns:A=”AirSyncBase:”>
     <xs:import namespace=”AirSyncBase” />
     <xs:element name="Anniversary" type="xs:dateTime" />
     <xs:element name="AssistantName" type="xs:string" />
     <xs:element name="AssistantPhoneNumber" type="xs:string" />
     <xs:element name="AssistnamePhoneNumber" type="xs:string" />
     <xs:element name="Birthday" type="xs:dateTime" />
     <xs:element name="Business2PhoneNumber" type="xs:string" />
     <xs:element name="BusinessAddressCity" type="xs:string" />
     <xs:element name="BusinessPhoneNumber" type="xs:string" />
     <xs:element name="WebPage" type="xs:string" />
     <xs:element name="BusinessAddressCountry" type="xs:string" />
     <xs:element name="Department" type="xs:string" />
     <xs:element name="Email1Address" type="xs:string" />
     <xs:element name="Email2Address" type="xs:string" />
     <xs:element name="Email3Address" type="xs:string" />
     <xs:element name="BusinessFaxNumber" type="xs:string" />
     <xs:element name="FileAs" type="xs:string" />
     <xs:element name="FirstName" type="xs:string" />
     <xs:element name="MiddleName" type="xs:string" />
     <xs:element name="HomeAddressCity" type="xs:string" />
     <xs:element name="HomeAddressCountry" type="xs:string" />
     <xs:element name="HomeFaxNumber" type="xs:string" />
     <xs:element name="HomePhoneNumber" type="xs:string" />
     <xs:element name="Home2PhoneNumber" type="xs:string" />
     <xs:element name="HomeAddressPostalCode" type="xs:string" />
     <xs:element name="HomeAddressState" type="xs:string" />
     <xs:element name="HomeAddressStreet" type="xs:string" />
     <xs:element name="MobilePhoneNumber" type="xs:string" />
     <xs:element name="Suffix" type="xs:string" />
     <xs:element name="CompanyName" type="xs:string" />
     <xs:element name="OtherAddressCity" type="xs:string" />
     <xs:element name="OtherAddressCountry" type="xs:string" />
     <xs:element name="CarPhoneNumber" type="xs:string" />
     <xs:element name="OtherAddressPostalCode" type="xs:string" />
     <xs:element name="OtherAddressState" type="xs:string" />
     <xs:element name="OtherAddressStreet" type="xs:string" />
     <xs:element name="PagerNumber" type="xs:string" />
     <xs:element name="Title" type="xs:string" />
     <xs:element name="BusinessPostalCode" type="xs:string" />
     <xs:element name="LastName" type="xs:string" />
     <xs:element name="Spouse" type="xs:string" />
     <xs:element name="BusinessState" type="xs:string" />
     <xs:element name="BusinessStreet" type="xs:string" />
     <xs:element name="JobTitle" type="xs:string" />
     <xs:element name="YomiFirstName" type="xs:string" />
     <xs:element name="YomiLastName" type="xs:string" />
     <xs:element name="YomiCompanyName" type="xs:string" />
     <xs:element name="OfficeLocation" type="xs:string" />
     <xs:element name="RadioPhoneNumber" type="xs:string" />
     <xs:element name ="Picture" type="xs:string"/>
     <xs:element name="Categories">
          <xs:complexType>
               <xs:sequence>
                    <xs:element name="Category" type="xs:string" />
               </xs:sequence>
          </xs:complexType>
     </xs:element>
     <xs:element name="Children">
          <xs:complexType>
               <xs:sequence>
                    <xs:element name="Child" type="xs:string" />
               </xs:sequence>
          </xs:complexType>
     </xs:element>
     <xs:element name="Body" type=”A:Body” />
</xs:schema>
```

The XML schema definition for the Contact2 class in ActiveSync is as follows:

```
<?xml version="1.0" ?>
<xs:schema xmlns:tns="CONTACTS2:" attributeFormDefault="unqualified"
elementFormDefault="qualified" targetNamespace="CONTACTS2:"
xmlns:xs=http://www.w3.org/2001/XMLSchema xmlns:A=”AirSyncBase:”>
     <xs:import namespace=”AirSyncBase” />
     <xs:element name ="CustomerId" type="xs:string" />
     <xs:element name ="GovernmentId" type="xs:string"/>
     <xs:element name ="IMAddress" type="xs:string"/>
     <xs:element name ="IMAddress2" type="xs:string"/>
     <xs:element name ="IMAddress3" type="xs:string"/>
     <xs:element name ="ManagerName" type="xs:string"/>
     <xs:element name ="CompanyMainPhone" type="xs:string"/>
     <xs:element name ="AccountName" type="xs:string" />
     <xs:element name ="NickName" type="xs:string" />
     <xs:element name ="MMS" type="xs:string"/>
</xs:schema>
```

2.2.1 Namespaces

This specification defines and references the following XML namespaces.

Prefix	Reference
CONTACTS:	[MS-ASCNTC]
CONTACTS2:	[MS-ASCNTC]
A:	[MS-ASAIRS]

2.2.2 Simple Types

This specification does not define any common XML schema simple types.

2.2.3 Complex Types

The following table summarizes the set of common XML schema complex type definitions defined by this specification.

Complex Type	Description
Categories	A collection of categories to which the contact belongs.
Children	A collection of the contact's children.
Body	The notes for this contact.

2.2.3.1 Categories

The **Categories** type is an optional **container** ([MS-ASDTYPE] section 2.8) type that specifies a collection of categories to which the contact belongs.

The **Categories** type MUST have the following child element:

- **Category** (section 2.2.4.60): At least one instance of this element is required.

2.2.3.2 Children

The **Children** type is an optional **container** ([MS-ASDTYPE] section 2.8) type that specifies a collection of the contact's children.

The **Children** type MUST have the following child element:

- **Child** (section 2.2.4.61): At least one instance of this element is required.

2.2.3.3 Body

The **Body** type is an optional container ([MS-ASDTYPE] section 2.8) type that specifies the notes for this contact. For more details about the **Body** type, see [MS-ASAIRS].

2.2.4 Elements

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

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

The following elements are defined in the Contacts namespace.

Element	Description
Anniversary	The wedding anniversary date for the contact.
AssistantName	The name of the contact's assistant.
AssistantPhoneNumber	The phone number of the contact's assistant.
Birthday	The birth date of the contact.
Business2PhoneNumber	The second business telephone number for the contact.
BusinessAddressCity	The business city of the contact.
BusinessPhoneNumber	The business telephone number for the contact.
WebPage	The Web site or personal Web page for the contact.
BusinessAddressCountry	The business country for the contact.
Department	The department name for the contact.
Email1Address	The first e-mail address for the contact.
Email2Address	The second e-mail address for the contact.
Email3Address	The third e-mail address for the contact.
BusinessFaxNumber	The business fax number for the contact.
FileAs	The filing string – and also, typically, the display name – for the contact.
FirstName	The contact's first name.
MiddleName	The contact's middle name.
HomeAddressCity	The home city for the contact.
HomeAddressCountry	The home country for the contact.

HomeFaxNumber	The home fax number for the contact.
HomePhoneNumber	The home phone number for the contact.
Home2PhoneNumber	The second home phone number for the contact.
HomeAddressPostalCode	The home postal code for the contact.
HomeAddressState	The home state for the contact.
HomeAddressStreet	The home street address for the contact.
MobilePhoneNumber	The mobile phone number for the contact.
Suffix	The suffix for the contact’s name.
CompanyName	The company name for the contact.
OtherAddressCity	The city of the contact’s alternate address.
OtherAddressCountry	The country of the contact’s alternate address.
CarPhoneNumber	The car telephone number for the contact.
OtherAddressPostalCode	The postal code of the contact’s alternate address.
OtherAddressState	The state of the contact’s alternate address.
OtherAddressStreet	The street address of the contact’s alternate address.
PagerNumber	The pager number for the contact.
Title	The contact’s business title.
BusinessPostalCode	The business postal code for the contact.
LastName	The contact’s last name.
Spouse	The name of the contact’s spouse.
BusinessState	The business state for the contact.
BusinessStreet	The business street address for the contact.
JobTitle	The contact’s job title.
YomiFirstName	The Japanese phonetic rendering of the first name of the contact.
YomiLastName	The Japanese phonetic rendering of the last name of the contact.
YomiCompanyName	The Japanese phonetic rendering of the company name for the contact.

OfficeLocation	The office location for the contact.
RadioPhoneNumber	The radio telephone number for the contact.
Picture	The Base64-encoded .jpg file containing the picture of the contact.
Categories.Category	A category to which the contact is assigned.
Children.Child	One of the child contacts underneath this contact.

The following elements are defined in the Contacts2 namespace.

Element	Description
CustomerId	The customer identifier (ID) for the contact.
GovernmentId	The government-assigned identifier (ID) for the contact.
IMAddress	The instant messaging address for the contact.
IMAddress2	The alternate instant messaging address for the contact.
IMAddress3	The tertiary instant messaging address for the contact.
ManagerName	The distinguished name (DN) of the manager for the contact.
CompanyMainPhone	The main telephone number for the contact’s company.
AccountName	The account name and/or number for the contact.
NickName	The nickname for the contact.
MMS	The Multimedia Messaging Service (MMS) address for the contact.

2.2.4.1 Anniversary

The **Anniversary** element is an optional element that specifies the wedding anniversary date for the contact. It is defined as an element in the Contacts namespace.

2.2.4.2 AssistantName

The **AssistantName** element is an optional element that specifies the name of the contact’s assistant. It is defined as an element in the Contacts namespace.

2.2.4.3 AssistantPhoneNumber

The **AssistantPhoneNumber** element is an optional element that specifies the phone number of the contact's assistant. It is defined as an element in the Contacts namespace.

2.2.4.4 Birthday

The **Birthday** element is an option element that specifies the birth date of the contact. It is defined as an element in the Contacts namespace.

2.2.4.5 Business2PhoneNumber

The **Business2PhoneNumber** element is an optional element that specifies the secondary business telephone number for the contact. It is defined as an element in the Contacts namespace.

2.2.4.6 BusinessAddressCity

The **BusinessAddressCity** element is an optional element that specifies the business city of the contact. It is defined as an element in the Contacts namespace.

2.2.4.7 BusinessPhoneNumber

The **BusinessPhoneNumber** element is an optional element that specifies the primary business phone number for the contact. It is defined as an element in the Contacts namespace.

2.2.4.8 WebPage

The **WebPage** element is an optional element that specifies the Web site or personal Web page for the contact. It is defined as an element in the Contacts namespace.

2.2.4.9 BusinessAddressCountry

The **BusinessAddressCountry** element is an optional element that specifies the business country of the contact. It is defined as an element in the Contacts namespace.

2.2.4.10 Department

The **Department** element is an optional element that specifies the department name for the contact. It is defined as an element in the Contacts namespace.

2.2.4.11 Email1Address

The **Email1Address** element is an optional element that specifies the first e-mail address for the contact. It is defined as an element in the Contacts namespace.

2.2.4.12 Email2Address

The **Email2Address** element is an optional element that specifies the second e-mail address for the contact. It is defined as an element in the Contacts namespace.

2.2.4.13 Email3Address

The **Email3Address** element is an optional element that specifies the third e-mail address for the contact. It is defined as an element in the Contacts namespace.

2.2.4.14 BusinessFaxNumber

The **BusinessFaxNumber** element is an optional element that specifies the business fax number for the contact. It is defined as an element in the Contacts namespace.

2.2.4.15 FileAs

The **FileAs** element is an optional element that specifies the filing name for the contact. It is defined as an element in the Contacts namespace.

2.2.4.16 FirstName

The **FirstName** element is a required element that specifies the first name of the contact. It is defined as an element in the Contacts namespace.

2.2.4.17 MiddleName

The **MiddleName** element is an optional element that specifies the middle name of the contact. It is defined as an element in the Contacts namespace.

2.2.4.18 HomeAddressCity

The **HomeAddressCity** element is an optional element that specifies the home city for the contact. It is defined as an element in the Contacts namespace.

2.2.4.19 HomeAddressCountry

The **HomeAddressCountry** element is an optional element that specifies the home country for the contact. It is defined as an element in the Contacts namespace.

2.2.4.20 HomeFaxNumber

The **HomeFaxNumber** element is an optional element that specifies the home fax number for the contact. It is defined as an element in the Contacts namespace.

2.2.4.21 HomePhoneNumber

TheHomePhoneNumber element is an optional element that specifies the home phone number for the contact. It is defined as an element in the Contacts namespace.

2.2.4.22 Home2PhoneNumber

The **Home2PhoneNumber** element is an optional element that specifies the alternate home phone number for the contact. It is defined as an element in the Contacts namespace.

2.2.4.23 HomeAddressPostalCode

The **HomeAddressPostalCode** element is an optional element that specifies the home postal code for the contact. It is defined as an element in the Contacts namespace.

2.2.4.24 HomeAddressState

The **HomeAddressState** element is an optional element that specifies the home state for the contact. It is defined as an element in the Contacts namespace.

2.2.4.25 HomeAddressStreet

The **HomeAddressStreet** element is an optional element that specifies the home street address for the contact. It is defined as an element in the Contacts namespace.

2.2.4.26 MobilePhoneNumber

The **MobilePhoneNumber** element is an optional element that specifies the mobile phone number for the contact. It is defined as an element in the Contacts namespace.

2.2.4.27 Suffix

The **Suffix** element is an optional element that specifies the suffix for the contact's name. It is defined as an element in the Contacts namespace.

2.2.4.28 CompanyName

The **CompanyName** element is an optional element that specifies the company name for the contact. It is defined as an element in the Contacts namespace.

2.2.4.29 OtherAddressCity

The **OtherAddressCity** element is an optional element that specifies the city for the contact's alternate address. It is defined as an element in the Contacts namespace.

2.2.4.30 OtherAddressCountry

The **OtherAddressCountry** element is an optional element that specifies the country of the contact's alternate address. It is defined as an element in the Contacts namespace.

2.2.4.31 CarPhoneNumber

The **CarPhoneNumber** element is an optional element that specifies the car telephone number for the contact. It is defined as an element in the Contacts namespace.

2.2.4.32 OtherAddressPostalCode

The **OtherAddressPostalCode** element is an optional element that specifies the postal code of the contact's alternate address. It is defined as an element in the Contacts namespace.

2.2.4.33 OtherAddressState

The **OtherAddressState** element is an optional element that specifies the state of the contact's alternate address. It is defined as an element in the Contacts namespace.

2.2.4.34 OtherAddressStreet

The **OtherAddressStreet** element is an optional element that specifies the street address of the contact's alternate address. It is defined as an element in the Contacts namespace.

2.2.4.35 PagerNumber

The **PagerNumber** element is an optional element that specifies the pager number for the contact. It is defined as an element in the Contacts namespace.

2.2.4.36 Title

The **Title** element is an optional element that specifies the contact's business title. It is defined as an element in the Contacts namespace.

2.2.4.37 BusinessPostalCode

The **BusinessPostalCode** is an optional element that specifies the business postal code for the contact. It is defined as an element in the Contacts namespace.

2.2.4.38 LastName

The **LastName** element is a required element that specifies the contact's last name. It is defined as an element in the Contacts namespace.

2.2.4.39 Spouse

The **Spouse** element is an optional element that specifies the name of the contact's spouse. It is defined as an element in the Contacts namespace.

2.2.4.40 BusinessState

The **BusinessState** element is an optional element that specifies the business state for the contact. It is defined as an element in the Contacts namespace.

2.2.4.41 BusinessStreet

The **BusinessStreet** element is an optional element that specifies the business street address for the contact. It is defined as an element in the Contacts namespace.

2.2.4.42 JobTitle

The **JobTitle** element is an optional element that specifies the contact's job title. It is defined as an element in the Contacts namespace.

2.2.4.43 YomiFirstName

The **YomiFirstName** element is an optional element that specifies the Japanese phonetic rendering of the first name of the contact. It is defined as an element in the Contacts namespace.

2.2.4.44 YomiLastName

The **YomiLastName** element is an optional element that specifies the Japanese phonetic rendering of the last name of the contact. It is defined as an element in the Contacts namespace.

2.2.4.45 YomiCompanyName

The **YomiCompanyName** element is an optional element that specifies the Japanese phonetic rendering of the company name for the contact. It is defined as an element in the Contacts namespace.

2.2.4.46 OfficeLocation

The **OfficeLocation** element is an optional element that specifies the office location for the contact. It is defined as an element in the Contacts namespace.

2.2.4.47 RadioPhoneNumber

The **RadioPhoneNumber** element is an optional element that specifies the radio phone number for the contact. It is defined as an element in the Contacts namespace.

2.2.4.48 CustomerId

The **CustomerId** element is an optional element that specifies the customer identifier (ID) for the contact. It is defined as an element in the Contacts2 namespace.

2.2.4.49 GovernmentId

The **GovernmentId** element is an optional element that specifies the government-assigned identifier (ID) for the contact. It is defined as an element in the Contacts2 namespace.

2.2.4.50 IMAddress

The **IMAddress** element is an optional element that specifies the instant messaging address for the contact. It is defined as an element in the Contacts2 namespace.

2.2.4.51 IMAddress2

The **IMAddress2** element is an optional element that specifies the alternate instant messaging address for the contact. It is defined as an element in the Contacts2 namespace.

2.2.4.52 IMAddress3

The **IMAddress3** element is an optional element that specifies the tertiary instant messaging address for the contact. It is defined as an element in the Contacts2 namespace.

2.2.4.53 ManagerName

The **ManagerName** element is an optional element that specifies the distinguished name (DN) of the contact's manager. It is defined as an element in the Contacts2 namespace.

2.2.4.54 CompanyMainPhone

The **CompanyMainPhone** element is an optional element that specifies the main telephone number for the contact's company. It is defined as an element in the Contacts2 namespace.

2.2.4.55 AccountName

The **AccountName** element is an optional element that specifies the account name and/or number for the contact. It is defined as an element in the Contacts2 namespace.

2.2.4.56 NickName

The **NickName** element is an optional element that specifies the nickname for the contact. It is defined as an element in the Contacts2 namespace.

2.2.4.57 MMS

The **MMS** element is an optional element that specifies the Multimedia Messaging Service (MMS) address for the contact. It is defined as an element in the Contacts2 namespace.

2.2.4.58 Picture

The **Picture** element is an optional element that specifies the Base64-encoded .jpg file containing the picture of the contact. It is defined as an element in the Contacts namespace.

The size of the value of **Picture** is limited to 36.6 kilobytes (KB).

2.2.4.59 Categories.Category

The **Categories.Category** element is an optional child element of the **Categories** type that specifies a category to which the contact is assigned. It is defined as an element in the Contacts namespace.

A command request or response has a minimum of one **Category** element per **Categories** type. It can have an unlimited number of Category elements per **Categories** type.

2.2.4.60 Children.Child

The **Children.Child** element is an optional child element of the **Children** type that specifies one of the child contacts underneath this contact. It is defined as an element in the Contacts namespace.

A command request or response has a minimum of one **Child** element per **Children** type. It can have an unlimited number of Category elements per **Children** type.

2.2.5 Attributes

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

2.2.6 Groups

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

2.2.7 Attribute Groups

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

2.2.8 Messages

Commands that send and receive Contact class data are specified in section 3.1.5.

3 Protocol Details

3.1 Client and Server 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.

The Contacts class is a structured **XML** text block that adheres to the XML schema definition specified in section 2.1. It is returned by the server as part of a full XML response to the client commands specified in section 3.1.4. It is processed by the client as part of the full XML body transmitted by the server in response to the client commands specified in section 3.1.4.

The server can return zero or more Contact class blocks in its response, depending on how many contacts match the criteria specified by the client command request. The server returns a Contacts class XML block for every contact that matches the criteria specified by the client command request.

The client can expect to process zero or more Contact class blocks in response to the client commands specified in section 3.1.4.

3.1.2 Timers

No timer actions are defined for the Contacts or Contacts2 class.

3.1.3 Initialization

None.

3.1.4 Higher-Layer Triggered Events

The Contact **class** is utilized when a client:

- Synchronizes its contact data with a server
- Searches a server for a contact
- Requests details for one or more specific contacts

3.1.5 Message Processing Events and Sequencing Rules

The following section describes how various elements of the Contacts class are used in the context of specific ActiveSync commands. Command details are specified in [MS-ASCMD].

3.1.5.1 ItemOperations

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

ItemOperations is specified in [MS-ASCMD] section 2.2.1.9.

3.1.5.1.1 Complex Types

3.1.5.1.1.1 Command Request

Any of the complex types for the Contact **class** can be included in an **ItemOperations** command request.

Contact **class** complex types are transmitted as children of the **Schema** type ([MS-ASCMD] section 2.2.1.9.2.12).

3.1.5.1.1.2 Command Response

Any of the complex types for the Contact **class** can be included in an **ItemOperations** command response. If a **Schema** element was included in the command request, then the complex types returned MUST be restricted to the complex types included in the command request's **Schema** element.

Contact **class** complex types are returned as children of the **Properties** type ([MS-ASCMD] section 2.2.1.9.3.8).

3.1.5.1.2 Elements

3.1.5.1.2.1 Command Request

Any of the Contact **class** elements can be included in an **ItemOperations** command request.

Contact **class** elements are transmitted as children of the **Schema** type ([MS-ASCMD] section 2.2.1.9.2.12)

3.1.5.1.2.2 Command Response

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

Contact **class** elements are returned as children of the **Schema** type ([MS-ASCMD] section 2.2.1.9.3.8).

3.1.5.2 Search

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

Search is specified in [MS-ASCMD] Section 2.2.1.16.

3.1.5.2.1 Complex Types

3.1.5.2.1.1 Command Request

The complex types for the Contact class MUST NOT be included in a **Search** command request.

3.1.5.2.1.2 Command Response

Any of the complex types for the Contact **class** can be included in a **Search** command response.

Contact **class** complex types are returned as children of the **Properties** type ([MS-ASCMD] section 2.2.1.16.2.2).

3.1.5.2.2 Elements

3.1.5.2.2.1 Command Request

The Contact **class** elements MUST NOT be included in a **Search** command request.

3.1.5.2.2.2 Command Response

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

Contact **class** elements are returned as children of the **Properties** type ([MS-ASCMD] section 2.2.1.16.2.2).

3.1.5.3 Sync

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

Sync is specified in [MS-ASCMD] section 2.2.1.21.

3.1.5.3.1 Complex Types

3.1.5.3.1.1 Command Request

Any of the complex types for the Contact **class** can be included in a **Sync** command request.

Contact **class** complex types are transmitted as children of the **ApplicationData** type ([MS-ASCMD] section 2.2.1.21.1.6). They can also be transmitted as children of the **Supported** type ([MS-ASCMD] section 2.2.1.21.1.13).

3.1.5.3.1.1.1 Body

A client can specify that the Notes contained in a Body text be truncated by sending a **Truncated** element to the server in a **Sync** command. The behavior of **Truncated** is specified in ([MS-ASAIRS] section 2.2.4.12).

Once a client device has requested that the **Notes** field be synchronized using truncation, it cannot request the entire **Notes** field again.

If a **Body** type is not sent from the client to the server, then the server MUST NOT delete the stored Notes for the contact.

Client devices that do not support Notes functionality for contacts MUST NOT send a **Body** type when synchronizing contact information with a server.

3.1.5.3.1.2 Command Response

Any of the complex types for the Contact **class** can be included in a **Sync** command response. If a **Supported** element was included in the command request, then the complex types returned MUST be restricted to the complex types included in the command request's **Supported** element.

Contact **class** complex types are returned as children of the **ApplicationData** type ([MS-ASCMD] section2.2.1.21.2.2).

3.1.5.3.2 Elements

3.1.5.3.2.1 Command Request

Any of the Contact **class** elements can be included in a **Sync** command request.

Contact class elements are transmitted as children of the **ApplicationData** type ([MS-ASCMD] section 2.2.1.21.1.6). They can also be transmitted as children of the **Supported** type ([MS-ASCMD] section 2.2.1.21.1.13).

3.1.5.3.2.2 Command Response

Any of the elements for the Contact **class** can be included in a **Sync** command response. If a **Supported** element was included in the command request, then the elements returned MUST be restricted to the elements included in the command request's **Properties** element.

Contact **class** elements are returned as children of the **ApplicationData** type ([MS-ASCMD] section 2.2.1.21.2.2).

3.1.6 Timer Events

None.

3.1.7 Other Local Events

None.

4 Protocol Examples

The following request/response transaction demonstrates a client requesting contact synchronization with a server, and the server returning a single new contact. In this example, the complex types and elements of the Contact **class** are contained as children of the **Add** type in the server response.

Note: In this example, the value of the **Picture** element, which is a Base64-encoded representation of the image, has been truncated.

Request:

```
<?xml version="1.0" encoding="utf-8"?>
<Sync xmlns="AirSync:">
  <Collections>
    <Collection>
      <SyncKey>2006814013</SyncKey>
      <CollectionId>2</CollectionId>
      <DeletesAsMoves/>
      <GetChanges/>
    </Collection>
  </Collections>
</Sync>
```

Response:

```
<?xml version="1.0" encoding="utf-8"?>
<Sync xmlns="AirSync:" xmlns:A="AirSyncBase:" xmlns:B="POOMCONTACTS:">
  <Collections>
    <Collection>
      <SyncKey>243360144</SyncKey>
      <CollectionId>2</CollectionId>
      <Status>1</Status>
      <Commands>
        <Add>
          <ServerId>2:1</ServerId>
          <ApplicationData>
            <A:Body>
              <A:Type>3</A:Type>
              <A:EstimatedDataSize>5500</A:EstimatedDataSize>
              <A:Truncated>1</A:Truncated>
            </A:Body>
            <B:WebPage>http://www.contoso.com/</B:WebPage>
            <B:BusinessCountry>United States of America</B:BusinessCountry>
            <B:Email1Address>"Rajesh M. Patel (someone@example.com)"
&lt;someone@example.com&gt;</B:Email1Address>
            <B:BusinessFaxNumber>(206) 555-7912</B:BusinessFaxNumber>
            <B:FileAs>Patel, Rajesh M.</B:FileAs>
            <B:FirstName>Rajesh</B:FirstName>
            <B:HomePhoneNumber>(206) 555-9201</B:HomePhoneNumber>
            <B:BusinessCity>Redmond</B:BusinessCity>
            <B:MiddleName>M.</B:MiddleName>
            <B:MobilePhoneNumber>(206) 555-3248</B:MobilePhoneNumber>
            <B:CompanyName>Contoso, Ltd.</B:CompanyName>
            <B:BusinessPostalCode>10021</B:BusinessPostalCode>
            <B:LastName>Patel</B:LastName>
            <B:BusinessState>WA</B:BusinessState>
            <B:BusinessStreet>234 Main St.</B:BusinessStreet>
            <B:BusinessPhoneNumber>(206) 555-9102</B:BusinessPhoneNumber>
            <B:JobTitle>Development Manager</B:JobTitle>
            <B:Picture>/9j/4AAQSkZJRgABAQEAYABgAAD/…</B:Picture>
            <A:NativeBodyType>3</A:NativeBodyType>
          </ApplicationData>
        </Add>
      </Commands>
    </Collection>
  </Collections>
</Sync>
```

5 Security

5.1 Security Considerations for Implementers

None.

5.2 Index of Security Parameters

None.

6 Appendix A: Office/Exchange Behavior

The information in this specification is applicable to the following versions of Office/Exchange:

- Office 2003 with Service Pack 3 applied
- Exchange 2003 with Service Pack 2 applied
- Office 2007 with Service Pack 1 applied
- Exchange 2007 with Service Pack 1 applied

Exceptions, if any, are noted below. Unless otherwise specified, any statement of optional behavior in this specification prescribed using the terms SHOULD or SHOULD NOT implies Office/Exchange behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies Office/Exchange does not follow the prescription.

Index