

[MS-OXWSCVTID]: Convert Item Identifier Web Service Protocol Specification

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft's Open Specification Promise (available here: <http://www.microsoft.com/interop/osp>) or the Community Promise (available here: <http://www.microsoft.com/interop/cp/default.msp>). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting 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
07/15/2009	1.0	Major	Initial Availability.
11/04/2009	1.1.0	Minor	Updated the technical content.
02/10/2010	1.2.0	Minor	Updated the technical content.
05/05/2010	1.2.1	Editorial	Revised and edited the technical content.
08/04/2010	1.3	Minor	Clarified the meaning of the technical content.
11/03/2010	1.4	Minor	Clarified the meaning of the technical content.

Contents

1 Introduction	5
1.1 Glossary	5
1.2 References	5
1.2.1 Normative References	5
1.2.2 Informative References	6
1.3 Overview	6
1.4 Relationship to Other Protocols	6
1.5 Prerequisites/Preconditions	6
1.6 Applicability Statement	7
1.7 Versioning and Capability Negotiation	7
1.8 Vendor-Extensible Fields	7
1.9 Standards Assignments	7
2 Messages	8
2.1 Transport	8
2.2 Common Message Syntax	8
2.2.1 Namespaces	8
2.2.2 Simple Types	8
2.2.3 Complex Types	8
2.2.4 Elements	8
2.2.5 Attributes	8
2.2.6 Groups	8
2.2.7 Attribute Groups	9
2.2.8 Message Syntax	9
3 Protocol Details	10
3.1 ExchangeServicePortType Server Details	10
3.1.1 Server Abstract Data Model	10
3.1.2 Server Timers	10
3.1.3 Server Initialization	10
3.1.4 Server Message Processing Events and Sequencing	10
3.1.4.1 ConvertId	10
3.1.4.1.1 ConvertId Simple Types	11
3.1.4.1.1.1 t:IdFormatType Simple Type	11
3.1.4.1.2 ConvertId Complex Types	12
3.1.4.1.2.1 m:ConvertIdResponseMessageType Complex Type	12
3.1.4.1.2.2 m:ConvertIdResponseType Complex Type	12
3.1.4.1.2.3 m:ConvertIdType Complex Type	13
3.1.4.1.2.4 t:AlternateIdBaseType Complex Type	13
3.1.4.1.2.5 t:AlternateIdType Complex Type	14
3.1.4.1.2.6 t:AlternatePublicFolderIdType Complex Type	15
3.1.4.1.2.7 t:AlternatePublicFolderItemIdType Complex Type	15
3.1.4.1.2.8 t:NonEmptyArrayOfAlternateIdsType Complex Type	16
3.1.4.1.3 ConvertId Elements	16
3.1.4.1.3.1 ConvertId Element	16
3.1.4.1.3.2 ConvertIdResponse Element	16
3.1.4.1.4 ConvertID Messages	17
3.1.4.1.4.1 tns:ConvertIdSoapIn	17
3.1.4.1.4.2 tns:ConvertIdSoapOut	17
3.1.5 Server Timer Events	17

3.1.6	Server Other Local Events	17
3.2	Client Details.....	17
3.2.1	Client Abstract Data Model	17
3.2.2	Client Timers	17
3.2.3	Client Initialization	18
3.2.4	Client Message Processing Events and Sequencing	18
3.2.5	Client Timer Events	18
3.2.6	Client Other Local Events	18
4	Protocol Examples.....	19
5	Security.....	20
5.1	Security Considerations for Implementers.....	20
5.2	Index of Security Parameters	20
6	Appendix A: Full WSDL.....	21
6.1	WSDL.....	21
6.2	Messages Schema.....	22
6.3	Types Schema.....	23
7	Appendix A: Product Behavior.....	25
8	Change Tracking.....	26
9	Index	28

1 Introduction

This document specifies the Convert Item Identifier Web Service protocol, which enables a client to convert among the different identifier formats that can be used to locate items that are stored on the server.

1.1 Glossary

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

EntryId
Hypertext Transfer Protocol (HTTP)
Hypertext Transfer Protocol over Secure Socket Layers (HTTPS)
mailbox
public folder
SOAP body
SOAP fault
SOAP message
Web Services Description Language (WSDL)
WSDL message
WSDL port type
XML
XML schema

The following terms are specific to this document:

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

1.2 References

1.2.1 Normative References

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

[MS-OXPROPS] Microsoft Corporation, "[Exchange Server Protocols Master Property List](#)", April 2008.

[MS-OXWSCDATA] Microsoft Corporation, "[Common Web Service Data Types](#)", July 2009.

[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>

[SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", W3C Note, May 2000, <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

[WSDL] Christensen, E., Curbera, F., Meredith, G., and Weerawarana, S., "Web Services Description Language (WSDL) 1.1", W3C Note, March 2001, <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>

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

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

[XMLSCHEMA2] Biron, P., and Malhotra, A., Eds., "XML Schema Part 2: Datatypes", W3C Recommendation, May 2001, <http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/>

1.2.2 Informative References

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

1.3 Overview

The Convert Item Identifier Web Service protocol enables clients to obtain alternative identifiers for items that are stored by the server. Those identifiers can then be used to access the stored items by using other protocols and programmatic interfaces that are available on the server.

1.4 Relationship to Other Protocols

The Convert Item Identifier Web Service protocol uses SOAP over **HTTP** and SOAP over **HTTPS**, as shown in the following figures.

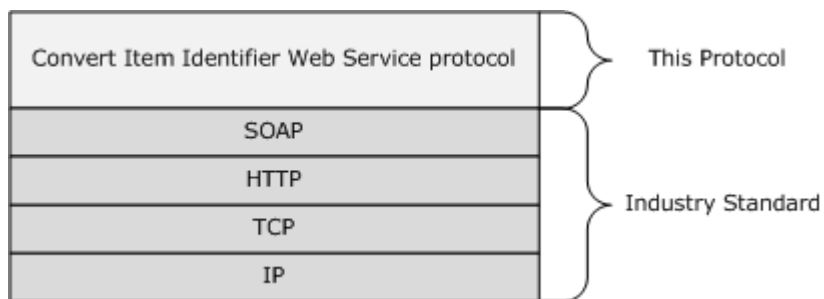


Figure 1: Convert Item Identifier Web Service protocol HTTP stack

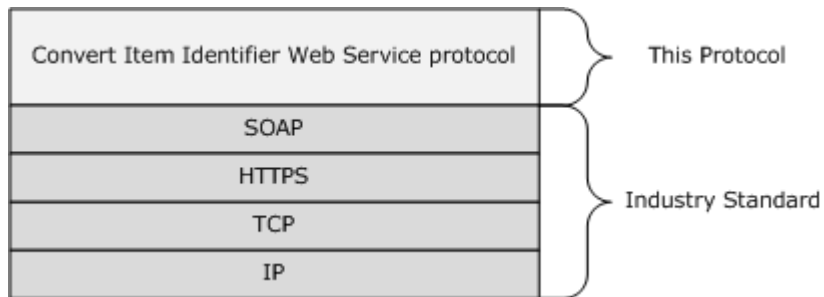


Figure 2: Convert Item Identifier Web Service protocol HTTPS stack

1.5 Prerequisites/Preconditions

None.

1.6 Applicability Statement

The Convert Item Identifier Web Service protocol is applicable to clients that must obtain alternative identifiers to an item that is stored on the server, and then use those identifiers to utilize other protocols or application programming interfaces to access the stored item.

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

- **Supported Transports:** This protocol uses SOAP 1.1, as specified in section [2.1](#).
- **Protocol Versions:** This protocol specifies only one **WSDL port type** version.
- **Security and Authentication Methods:** This protocol relies on the Web server that is hosting it to perform authentication.
- **Localization:** This protocol includes text strings in various messages. Localization considerations for such strings are specified in section [3.1.4](#).
- **Capability Negotiation:** None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

The SOAP version supported is SOAP 1.1. For details, see [\[SOAP1.1\]](#).

2.2 Common Message Syntax

This section contains common definitions that are used by this protocol. The syntax of the definitions uses **XML schema** as defined in [\[XMLSCHEMA1\]](#) and [\[XMLSCHEMA2\]](#), and **Web Services Description Language (WSDL)** as defined in [\[WSDL\]](#).

2.2.1 Namespaces

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

Prefix	Namespace URI	Reference
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
tns	http://schemas.microsoft.com/exchange/services/2006/messages	[MS-OXWSCVTID]
s	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1]
targetNamespace	http://schemas.microsoft.com/exchange/services/2006/messages	[MS-OXWSCVTID]
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
t	http://schemas.microsoft.com/exchange/services/2006/types	[MS-OXWSCVTID]

2.2.2 Simple Types

This specification does not define any common XML schema simple type definitions.

2.2.3 Complex Types

This specification does not define any common XML schema Complex Type definitions.

2.2.4 Elements

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

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 Message Syntax

This specification does not define any common XML schema Message Syntax definitions.

3 Protocol Details

The client side of this protocol is simply a pass-through. That is, no additional timers or other state is required on the client side of this protocol. Calls made by the higher-layer protocol or application are passed directly to the transport, and the results that are returned by the transport are passed directly back to the higher-layer protocol or application.

3.1 ExchangeServicePortType Server Details

The Convert Item Identifier Web Service protocol defines a single port type with one operation.

Operation	Description
ConvertId	Converts an item identifier into a different identifier format that can be used to refer to the item. The ConvertId operation is defined in section 3.1.4.1 .

3.1.1 Server Abstract Data Model

The Convert Item Identifier Web Service protocol is a stateless protocol.

3.1.2 Server Timers

None.

3.1.3 Server Initialization

None.

3.1.4 Server Message Processing Events and Sequencing

This protocol includes the operation that is listed in the following table.

Operation	Description
ConvertId	Converts the supplied item identifier into a different format. This operation is defined in section 3.1.4.1 .

3.1.4.1 ConvertId

The **ConvertID** operation converts the item identifier supplied by the client into a different identifier format. [<1>](#)

```
<wsdl:operation name="ConvertId">
  <wsdl:input message="tns:ConvertIdSoapIn"/>
  <wsdl:output message="tns:ConvertIdSoapOut"/>
</wsdl:operation>
```

Request

Message Format	Description
tns:ConvertIdSoapIn	Specifies the SOAP message that requests the server time zone.

Response

Message Format	Description
tns:ConvertIdSoapOut	Specifies the SOAP message that is returned by the server in response.

3.1.4.1.1 ConvertId Simple Types

The following XML schema simple types are specific to this operation.

3.1.4.1.1.1 t:IdFormatType Simple Type

The **IdFormatType** simple type specifies the item identifier format in both the client request and for each returned identifier.

```
<xs:simpleType name="IdFormatType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="EwsLegacyId"
    />
    <xs:enumeration
      value="EwsId"
    />
    <xs:enumeration
      value="EntryId"
    />
    <xs:enumeration
      value="HexEntryId"
    />
    <xs:enumeration
      value="StoreId"
    />
    <xs:enumeration
      value="OwaId"
    />
  </xs:restriction>
</xs:simpleType>
```

Enumeration

The following values are defined by the **IdFormatType** simple type:

Value	Description
EwsLegacyId	Specifies that the identifier is in the format that is used by the legacy version of the protocol.
EwsId	Specifies that the identifier is in the format that is used by the current version of the protocol.
EntryId	Specifies that the identifier is in the format that can be used with the PidTagEntryId , as specified in [MS-OXPROPS] section 2.761.

Value	Description
HexEntryId	Specifies that the identifier is a hexadecimal-encoded representation of the format that can be used with the PidTagEntryId, as specified in [MS-OXPROPS] section 2.761
StoreId	Specifies that the identifier is in a format that is recognized only by the server.
OwaId	Specifies that the identifier is in a format that is used by the Web-based client capability of the server, if one exists. <2>

3.1.4.1.2 ConvertId Complex Types

The following XML schema complex types are specific to this operation.

3.1.4.1.2.1 m:ConvertIdResponseMessageType Complex Type

The **ConvertIdResponseMessageType** complex type specifies the status and result of a **ConvertId** request (section [3.1.4.1](#)). The **ConvertIdResponseMessageType** complex type extends the **ResponseMessageType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.3.52.

```
<xs:complexType name="ConvertIdResponseMessageType">
  <xs:complexContent>
    <xs:extension
      base="m:ResponseMessageType"
    >
      <xs:sequence>
        <xs:element name="AlternateId"
          type="t:AlternateIdBaseType"
          maxOccurs="1"
          minOccurs="0"
        />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Child Elements

Element	Type	Description
AlternateId	t:AlternateIdBaseType	Specifies a converted identifier in the response.

3.1.4.1.2.2 m:ConvertIdResponseType Complex Type

The **ConvertIdResponseType** complex type specifies the contents of a response to a **ConvertId** request (section [3.1.4.1](#)). The **ConvertIdResponseType** complex type extends the **BaseResponseMessageType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.3.15.

```
<xs:complexType name="ConvertIdResponseType">
  <xs:complexContent>
    <xs:extension
      base="m:BaseResponseMessageType"
    >

```

```

    />
  </xs:complexContent>
</xs:complexType>

```

3.1.4.1.2.3 m:ConvertIdType Complex Type

The **ConvertIdType** complex type specifies the contents of a request to a **ConvertId** request (section [3.1.4.1](#)). The **ConvertIdType** complex type extends the **BaseRequestType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.3.14.

```

<xs:complexType name="ConvertIdType">
  <xs:complexContent>
    <xs:extension
      base="BaseRequestType"
    >
      <xs:sequence>
        <xs:element name="SourceIds"
          type="t:NonEmptyArrayOfAlternateIdsType"
        />
      </xs:sequence>
      <xs:attribute name="DestinationFormat"
        type="IdFormatType"
        use="required"
      />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Child Elements

Element	Type
SourceIds	t:NonEmptyArrayOfAlternateIdsType

Attributes

Name	Type
DestinationFormat	IdFormatType

3.1.4.1.2.4 t:AlternateIdBaseType Complex Type

The **AlternateIdBaseType** complex type specifies the base container type for the attributes that specify information about the returned item identifier. **AlternateIdBaseType** does not extend any other complex type.

```

<xs:complexType name="AlternateIdBaseType"
  abstract="true"
>
  <xs:attribute name="Format"
    type="IdFormatType"
    use="required"
  />

```

```

/>
</xs:complexType>

```

Attributes

Name	Type
Format	IdFormatType

3.1.4.1.2.5 t:AlternateIdType Complex Type

The **AlternateIdType** complex type specifies the information that is provided with a returned **mailbox** folder or mailbox item identifier. The **AlternateIdType** complex type extends the **AlternateIdBaseType** complex type (section [3.1.4.1.2.4](#)).

```

<xs:complexType name="AlternateIdType">
  <xs:complexContent>
    <xs:extension
      base="t:AlternateIdBaseType"
    >
      <xs:attribute name="Id"
        type="xs:string"
        use="required"
      />
      <xs:attribute name="Mailbox"
        type="t:NonEmptyStringType"
        use="required"
      />
      <xs:attribute name="IsArchive"
        type="xs:boolean"
      />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Attributes

Name	Type	Description
Id	xs:string	Specifies the source identifier in a request and the destination identifier in a response. This attribute MUST be present.
Mailbox	t:NonEmptyStringType	Specifies the mailbox primary SMTP address of the identifier to convert. This attribute MUST be present. The NonEmptyStringType simple type is specified in [MS-OXWSCDATA] section 2.2.2.18.
IsArchive	xs:boolean	Specifies whether the identifier represents an archived item. This attribute MUST be present if the item is an archived item. Otherwise, this attribute is optional. If this attribute is not present, the value is assumed to equal false .<3>

3.1.4.1.2.6 t:AlternatePublicFolderIdType Complex Type

The **AlternatePublicFolderIdType** complex type specifies the information that is provided with a returned **public folder** item identifier. The **AlternatePublicFolderIdType** complex type extends the **AlternateIdBaseType** complex type (section [3.1.4.1.2.4](#)).

```
<xs:complexType name="AlternatePublicFolderIdType">
  <xs:complexContent>
    <xs:extension
      base="t:AlternateIdBaseType"
    >
      <xs:attribute name="FolderId"
        type="xs:string"
        use="required"
      />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Attributes

Name	Type
FolderId	xs:string

3.1.4.1.2.7 t:AlternatePublicFolderItemIdType Complex Type

The **AlternatePublicFolderItemIdType** complex type specifies the information that is provided with a returned item that is located in a public folder. The **AlternatePublicFolderItemIdType** complex type extends the **AlternatePublicFolderIdType** complex type (section [3.1.4.1.2.6](#)).

```
<xs:complexType name="AlternatePublicFolderItemIdType">
  <xs:complexContent>
    <xs:extension
      base="t:AlternatePublicFolderIdType"
    >
      <xs:attribute name="ItemId"
        type="xs:string"
        use="required"
      />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Attributes

Name	Type
ItemId	xs:string

3.1.4.1.2.8 t:NonEmptyArrayOfAlternateIdsType Complex Type

The **NonEmptyArrayOfAlternateIdsType** complex type specifies a container for one or more item identifiers. Each individual identifier MUST be an **AlternateIdType** (section [3.1.4.1.2.5](#)), **AlternatePublicFolderIdType** (section [3.1.4.1.2.6](#)), or **AlternatePublicFolderItemIdType** (section [3.1.4.1.2.7](#)) complex type. The **NonEmptyArrayOfAlternateIdsType** complex type does not extend any other complex type.

```
<xs:complexType name="NonEmptyArrayOfAlternateIdsType">
  <xs:choice
    maxOccurs="unbounded"
  >
    <xs:element name="AlternateId"
      type="t:AlternateIdType"
    />
    <xs:element name="AlternatePublicFolderId"
      type="t:AlternatePublicFolderIdType"
    />
    <xs:element name="AlternatePublicFolderItemId"
      type="t:AlternatePublicFolderItemIdType"
    />
  </xs:choice>
</xs:complexType>
```

Child Elements

Element	Type
AlternateId	t:AlternateIdType
AlternatePublicFolderId	t:AlternatePublicFolderIdType
AlternatePublicFolderItemId	t:AlternatePublicFolderItemIdType

3.1.4.1.3 ConvertId Elements

The following XML schema element definitions are specific to this operation.

3.1.4.1.3.1 ConvertId Element

The <ConvertId> element defines a request to convert a supplied list of source item identifiers into the requested destination format.

```
<xs:element name="ConvertId"
  type="m:ConvertIdType"
/>
```

3.1.4.1.3.2 ConvertIdResponse Element

The <ConvertIdResponse> element specifies a response from the server, and contains the alternate item identifiers for the supplied items in the requested format.


```
<xs:element name="ConvertIdResponse"
  type="m:ConvertIdResponseType"
/>
```

3.1.4.1.4 ConvertID Messages

The following **WSDL message** definitions are specific to this operation.

3.1.4.1.4.1 tns:ConvertIdSoapIn

The **ConvertIdSoapIn** message contains two parts, as described in the following table.

Part Name	Element/Type	Description
request	tns:ConvertId	Specifies the request.
RequestVersion	t:RequestServerVersion	Specifies the schema version for the request.

3.1.4.1.4.2 tns:ConvertIdSoapOut

The **ConvertIdSoapOut** message contains two parts, as described in the following table.

Part Name	Element/Type	Description
ConvertIdResult	tns:ConvertIdResponse	Specifies the response from the server.
ServerVersion	t:ServerVersionInfo	Specifies the server version for the response.

3.1.5 Server Timer Events

None.

3.1.6 Server Other Local Events

None.

3.2 Client Details

The client side of this protocol is simply a pass-through. That is, no additional timers or other state is required on the client side of this protocol. Calls made by the higher-layer protocol or application are passed directly to the transport, and the results returned by the transport are passed directly back to the higher-layer protocol or application.

3.2.1 Client Abstract Data Model

None.

3.2.2 Client Timers

None.

3.2.3 Client Initialization

None.

3.2.4 Client Message Processing Events and Sequencing

None.

3.2.5 Client Timer Events

None.

3.2.6 Client Other Local Events

None.

4 Protocol Examples

None.

5 Security

5.1 Security Considerations for Implementers

The Convert Item Identifier Web service protocol does not use any additional security mechanisms.

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

The following table lists the **XML** files that are required to implement the functionality that is specified in this document. The contents of each file are included in this section.

File name	Description	Section
MS-OXWSCVTID.wsdl	Contains the WSDL for the implementation of this protocol.	6.1
MS-OXWSCVTID-messages.xsd	Contains the XML schema message definitions that are used in this protocol.	6.2
MS-OXWSCVTID-types.xsd	Contains the XML schema type definitions that are used in this protocol.	6.3

These files have to be placed in a common folder for the WSDL to validate and operate. Also, any schema files that are included in or imported into the MS-OXWSCVTID-types.xsd or MS-OXWSCVTID-messages.xsd schemas have to be placed in the common folder along with the files.

6.1 WSDL

This section contains the contents of the MS-OXWSCVTID.wsdl file and information about additional files that this schema file requires to operate correctly.

MS-OXWSCVTID.wsdl includes the files listed in the following table. For the schema file to operate correctly, these files have to be present in the folder that contains the WSDL, types schema, and messages schema files for this protocol.

File name	Defining specification
MS-OXWSCVTID-messages.xsd	section 6.2 .
MS-OXWSCVTID-types.xsd	section 6.3

```
<?xml version="1.0" encoding="utf-8"?>

<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
  xmlns:s="http://www.w3.org/2001/XMLSchema" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
  targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages">
  <wsdl:types>
    <xs:schema id="messages" elementFormDefault="qualified" version="Exchange2010"
      xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
      xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
      xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
      xmlns:xs="http://www.w3.org/2001/XMLSchema"
      targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages">
      <xs:include schemaLocation="MS-OXWSCVTID-messages.xsd"/>
    </xs:schema>
    <xs:schema id="types" elementFormDefault="qualified" version="Exchange2010"
      xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
      targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types"
      xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/types"
      xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:include schemaLocation="MS-OXWSCVTID-types.xsd"/>
    </xs:schema>
  </wsdl:types>
</wsdl:definitions>
```

```

</wsdl:types>
<wsdl:message name="ConvertIdSoapIn" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
<wsdl:part name="request" element="tns:ConvertId"/>
<wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
</wsdl:message>
<wsdl:message name="ConvertIdSoapOut" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
<wsdl:part name="ConvertIdResult" element="tns:ConvertIdResponse"/>
<wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
<wsdl:portType name="ExchangeServicePortType">
<wsdl:operation name="ConvertId" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
<wsdl:input message="tns:ConvertIdSoapIn"/>
<wsdl:output message="tns:ConvertIdSoapOut"/>
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="ExchangeServiceBinding" type="tns:ExchangeServicePortType">
<soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
<wsdl:operation name="ConvertId" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
<soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/ConvertId"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
<wsdl:input>
<soap:header message="tns:ConvertIdSoapIn" part="RequestVersion" use="literal"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
<soap:body parts="request" use="literal" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
</wsdl:input>
<wsdl:output>
<soap:body parts="ConvertIdResult" use="literal"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
<soap:header message="tns:ConvertIdSoapOut" part="ServerVersion" use="literal"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
</wsdl:definitions>

```

6.2 Messages Schema

This section contains the contents of the MS-OXWSCVTID-messages.xsd file and information about additional files that this schema file requires to operate correctly.

MS-OXWSCVTID-messages.xsd includes or imports the files listed in the following table. For the schema file to operate correctly, these files have to be in the folder that contains the WSDL, types schema, and messages schema files for this protocol.

File name	Defining specification
MS-OXWSCDATA-messages.xsd	[MS-OXWSCDATA] section 6.2
MS-OXWSCVTID-types.xsd	Section 6.3

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<xs:schema xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
```

```

xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
elementFormDefault="qualified" version="Exchange2010" id="messages">
  <xs:import namespace="http://schemas.microsoft.com/exchange/services/2006/types"
schemaLocation="MS-OXWSCVTID-types.xsd"/>
  <xs:include schemaLocation="MS-OXWSCDATA-messages.xsd"/>
  <xs:complexType name="ConvertIdResponseMessageType"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:complexContent>
      <xs:extension base="m:ResponseMessageType">
        <xs:sequence>
          <xs:element name="AlternateId" type="t:AlternateIdBaseType"
minOccurs="0" maxOccurs="1"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="ConvertIdResponseType"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:complexContent>
      <xs:extension base="m:BaseResponseMessageType"/>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="ConvertIdType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:complexContent>
      <xs:extension base="m:BaseRequestType">
        <xs:sequence>
          <xs:element name="SourceIds"
type="t:NonEmptyArrayOfAlternateIdsType" minOccurs="1" maxOccurs="1"/>
        </xs:sequence>
        <xs:attribute name="DestinationFormat" type="t:IdFormatType"
use="required"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="ConvertId" type="m:ConvertIdType"
xmlns:xs="http://www.w3.org/2001/XMLSchema"/>
  <xs:element name="ConvertIdResponse" type="m:ConvertIdResponseType"
xmlns:xs="http://www.w3.org/2001/XMLSchema"/>
</xs:schema>

```

6.3 Types Schema

This section contains the contents of the MS-OXWSCVTID-types.xsd file and information about additional files that this schema file requires to operate correctly.

MS-OXWSCVTID-types.xsd includes the file listed in the following table. For the schema file to operate correctly, this file has to be in the folder that contains the WSDL, types schema, and messages schema files for this protocol.

File name	Defining specification
MS-OXWSCDATA-types.xsd	[MS-OXWSCDATA] section 6.3

```

<?xml version="1.0" encoding="utf-8"?>

<xs:schema xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types"
elementFormDefault="qualified" version="Exchange2010" id="types">
<xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
<xs:include schemaLocation="MS-OXWSCDATA-types.xsd"/>
<xs:complexType name="AlternateIdBaseType" abstract="true"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:attribute name="Format" type="t:IdFormatType" use="required"/>
</xs:complexType>
<xs:complexType name="AlternateIdType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:complexContent>
<xs:extension base="t:AlternateIdBaseType">
<xs:attribute name="Id" type="xs:string" use="required"/>
<xs:attribute name="Mailbox" type="t:NonEmptyStringType" use="required"/>
<xs:attribute name="IsArchive" type="xs:boolean" use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="AlternatePublicFolderIdType"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:complexContent>
<xs:extension base="t:AlternateIdBaseType">
<xs:attribute name="FolderId" type="xs:string" use="required"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="AlternatePublicFolderItemIdType"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:complexContent>
<xs:extension base="t:AlternatePublicFolderIdType">
<xs:attribute name="ItemId" type="xs:string" use="required"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:simpleType name="IdFormatType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:restriction base="xs:string">
<xs:enumeration value="EwsLegacyId"/>
<xs:enumeration value="EwsId"/>
<xs:enumeration value="EntryId"/>
<xs:enumeration value="HexEntryId"/>
<xs:enumeration value="StoreId"/>
<xs:enumeration value="OwaId"/>
</xs:restriction>
</xs:simpleType>
<xs:complexType name="NonEmptyArrayOfAlternateIdsType"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:choice minOccurs="1" maxOccurs="unbounded">
<xs:element name="AlternateId" type="t:AlternateIdType"/>
<xs:element name="AlternatePublicFolderId" type="t:AlternatePublicFolderIdType"/>
<xs:element name="AlternatePublicFolderItemId" type="t:AlternatePublicFolderItemIdType"/>
</xs:choice>
</xs:complexType>
</xs:schema>

```


7 Appendix A: Product Behavior

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

- Microsoft® Exchange Server 2007 Service Pack 1 (SP1)
- Microsoft® Outlook® 2010
- Microsoft® Exchange Server 2010
- Microsoft® Exchange Server 2010 Service Pack 1 (SP1)

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

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

[<1> Section 3.1.4.1.1](#): Outlook 2010 uses the **ConvertID** operation to convert an **EntryId** to an **EwsId**.

[<2> Section 3.1.4.1.1.1](#): Exchange 2007 SP1 includes the Web-based client Microsoft Office Outlook Web Access and Exchange 2010 includes the Web-based client Microsoft Office Outlook Web App. Identifiers in the OwaId format can be used to retrieve items.

[<3> Section 3.1.4.1.2.5](#): Exchange 2007 SP1 and Exchange 2010 do not include the **IsArchive** attribute. The **xs:boolean** type is specified in [\[XMLSCHEMA2\]](#).

8 Change Tracking

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change Type
Z Appendix A: Product Behavior	58043 Updated applicable product with Exchange Server 2007 SP1.	N	Content updated.

9 Index

A

[Applicability](#) 7

C

[Capability negotiation](#) 7

[Change tracking](#) 26

Client

[abstract data model](#) 17

[initialization](#) 18

[local events](#) 18

[message processing](#) 18

[overview](#) 17

[sequencing rules](#) 18

[timer events](#) 18

[timers](#) 17

F

[Full WSDL](#) 21

G

[Glossary](#) 5

I

[Introduction](#) 5

M

Messages

[overview](#) 8

[syntax](#) 8

[transport](#) 8

O

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

P

[Preconditions](#) 6

[Prerequisites](#) 6

[Product behavior](#) 25

R

References

[informative](#) 6

[normative](#) 5

[Relationship to other protocols](#) 6

S

Security

[implementer considerations](#) 20

[overview](#) 20

[parameter index](#) 20

Server

[abstract data model](#) 10

[initialization](#) 10

[local events](#) 17

[message processing](#) 10

[overview](#) 10

[sequencing rules](#) 10

[timer events](#) 17

[timers](#) 10

[Standards assignments](#) 7

T

[Tracking changes](#) 26

V

[Vendor-extensible fields](#) 7

[Versioning](#) 7