

# [MS-OXWSUSRCFG]:

## User Configuration Web Service Protocol

---

### Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation (“this documentation”) for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.
- **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 can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft [Open Specifications Promise](#) or the [Microsoft Community Promise](#). If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting [iplg@microsoft.com](mailto:iplg@microsoft.com).
- **License Programs.** To see all of the protocols in scope under a specific license program and the associated patents, visit the [Patent Map](#).
- **Trademarks.** The names of companies and products contained in this documentation might be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit [www.microsoft.com/trademarks](http://www.microsoft.com/trademarks).
- **Fictitious Names.** The example companies, organizations, products, domain names, email addresses, logos, people, places, and events that are 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 as specifically described above, whether by implication, estoppel, or otherwise.

**Tools.** The Open Specifications documentation does 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 documents are intended for use in conjunction with publicly available standards specifications and network programming art and, as such, assume that the reader either is familiar with the aforementioned material or has immediate access to it.

**Support.** For questions and support, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com).

## Revision Summary

Date	Revision History	Revision Class	Comments
7/15/2009	1.0	Major	Initial Availability.
11/4/2009	1.0.1	Editorial	Revised and edited the technical content.
2/10/2010	1.2.0	Minor	Updated the technical content.
5/5/2010	1.2.1	Editorial	Revised and edited the technical content.
8/4/2010	1.3	Minor	Clarified the meaning of the technical content.
11/3/2010	1.3	None	No changes to the meaning, language, or formatting of the technical content.
3/18/2011	2.0	Major	Significantly changed the technical content.
8/5/2011	3.0	Major	Significantly changed the technical content.
10/7/2011	3.0	None	No changes to the meaning, language, or formatting of the technical content.
1/20/2012	4.0	Major	Significantly changed the technical content.
4/27/2012	4.0	None	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	4.0	None	No changes to the meaning, language, or formatting of the technical content.
10/8/2012	4.1	Minor	Clarified the meaning of the technical content.
2/11/2013	4.1	None	No changes to the meaning, language, or formatting of the technical content.
7/26/2013	5.0	Major	Significantly changed the technical content.
11/18/2013	6.0	Major	Significantly changed the technical content.
2/10/2014	6.0	None	No changes to the meaning, language, or formatting of the technical content.
4/30/2014	7.0	Major	Significantly changed the technical content.
7/31/2014	7.0	None	No changes to the meaning, language, or formatting of the technical content.
10/30/2014	7.0.1	Editorial	Changed language and formatting in the technical content.
5/26/2015	8.0	Major	Significantly changed the technical content.
9/14/2015	8.0	None	No changes to the meaning, language, or formatting of the technical content.
6/13/2016	8.0	None	No changes to the meaning, language, or formatting of the technical content.
9/14/2016	8.0	None	No changes to the meaning, language, or formatting of the technical content.
6/20/2017	8.1	Minor	Clarified the meaning of the technical content.

<b>Date</b>	<b>Revision History</b>	<b>Revision Class</b>	<b>Comments</b>
7/24/2018	9.0	Major	Significantly changed the technical content.
10/1/2018	10.0	Major	Significantly changed the technical content.
12/11/2018	10.1	Minor	Clarified the meaning of the technical content.

# Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>7</b>
1.1	Glossary .....	7
1.2	References .....	8
1.2.1	Normative References .....	8
1.2.2	Informative References .....	9
1.3	Overview .....	9
1.4	Relationship to Other Protocols .....	9
1.5	Prerequisites/Preconditions .....	10
1.6	Applicability Statement .....	10
1.7	Versioning and Capability Negotiation .....	10
1.8	Vendor-Extensible Fields .....	10
1.9	Standards Assignments.....	10
<b>2</b>	<b>Messages.....</b>	<b>11</b>
2.1	Transport .....	11
2.2	Common Message Syntax .....	11
2.2.1	Namespaces .....	11
2.2.2	Messages.....	11
2.2.3	Elements .....	11
2.2.4	Complex Types.....	12
2.2.4.1	t:UserConfigurationDictionaryEntryType Complex Type .....	12
2.2.4.2	t:UserConfigurationDictionaryObjectType Complex Type .....	12
2.2.4.3	t:UserConfigurationDictionaryType Complex Type .....	13
2.2.4.4	t:UserConfigurationNameType Complex Type .....	13
2.2.4.5	t:UserConfigurationType Complex Type .....	14
2.2.5	Simple Types .....	15
2.2.5.1	t:UserConfigurationDictionaryObjectTypesType Simple Type .....	15
2.2.6	Attributes .....	16
2.2.7	Groups .....	16
2.2.8	Attribute Groups.....	16
<b>3</b>	<b>Protocol Details .....</b>	<b>17</b>
3.1	ExchangeServicePortType Server Details.....	17
3.1.1	Abstract Data Model.....	17
3.1.2	Timers .....	17
3.1.3	Initialization .....	17
3.1.4	Message Processing Events and Sequencing Rules .....	17
3.1.4.1	CreateUserConfiguration .....	17
3.1.4.1.1	Messages .....	18
3.1.4.1.1.1	CreateUserConfigurationSoapIn .....	18
3.1.4.1.1.2	CreateUserConfigurationSoapOut .....	19
3.1.4.1.2	Elements.....	20
3.1.4.1.2.1	CreateUserConfiguration Element.....	20
3.1.4.1.2.2	CreateUserConfigurationResponse Element .....	20
3.1.4.1.3	Complex Types .....	20
3.1.4.1.3.1	m:CreateUserConfigurationResponseType Complex Type .....	21
3.1.4.1.3.2	m:CreateUserConfigurationType Complex Type .....	21
3.1.4.1.4	Simple Types .....	21
3.1.4.1.5	Attributes .....	21
3.1.4.1.6	Groups .....	21
3.1.4.1.7	Attribute Groups.....	22
3.1.4.2	DeleteUserConfiguration .....	22
3.1.4.2.1	Messages .....	22
3.1.4.2.1.1	DeleteUserConfigurationSoapIn .....	22
3.1.4.2.1.2	DeleteUserConfigurationSoapOut .....	23

3.1.4.2.2	Elements .....	24
3.1.4.2.2.1	DeleteUserConfiguration Element .....	24
3.1.4.2.2.2	DeleteUserConfigurationResponse Element .....	24
3.1.4.2.3	Complex Types .....	24
3.1.4.2.3.1	m:DeleteUserConfigurationResponseType Complex Type .....	25
3.1.4.2.3.2	m:DeleteUserConfigurationType Complex Type .....	25
3.1.4.2.4	Simple Types .....	25
3.1.4.2.5	Attributes .....	26
3.1.4.2.6	Groups .....	26
3.1.4.2.7	Attribute Groups .....	26
3.1.4.3	GetUserConfiguration .....	26
3.1.4.3.1	Messages .....	26
3.1.4.3.1.1	GetUserConfigurationSoapIn .....	27
3.1.4.3.1.2	GetUserConfigurationSoapOut .....	27
3.1.4.3.2	Elements .....	28
3.1.4.3.2.1	GetUserConfiguration Element .....	28
3.1.4.3.2.2	GetUserConfigurationResponse Element .....	28
3.1.4.3.3	Complex Types .....	29
3.1.4.3.3.1	m:GetUserConfigurationResponseMessageType Complex Type .....	29
3.1.4.3.3.2	m:GetUserConfigurationType Complex Type .....	29
3.1.4.3.4	Simple Types .....	30
3.1.4.3.4.1	t:UserConfigurationPropertyType Simple Type .....	30
3.1.4.3.5	Attributes .....	31
3.1.4.3.6	Groups .....	31
3.1.4.3.7	Attribute Groups .....	31
3.1.4.4	UpdateUserConfiguration .....	31
3.1.4.4.1	Messages .....	32
3.1.4.4.1.1	UpdateUserConfigurationSoapIn .....	32
3.1.4.4.1.2	UpdateUserConfigurationSoapOut .....	33
3.1.4.4.2	Elements .....	34
3.1.4.4.2.1	UpdateUserConfiguration Element .....	34
3.1.4.4.2.2	UpdateUserConfigurationResponse Element .....	34
3.1.4.4.3	Complex Types .....	34
3.1.4.4.3.1	m:UpdateUserConfigurationResponseType Complex Type .....	34
3.1.4.4.3.2	m:UpdateUserConfigurationType Complex Type .....	35
3.1.4.4.4	Simple Types .....	35
3.1.4.4.5	Attributes .....	35
3.1.4.4.6	Groups .....	35
3.1.4.4.7	Attribute Groups .....	35
3.1.5	Timer Events .....	35
3.1.6	Other Local Events .....	36
<b>4</b>	<b>Protocol Examples .....</b>	<b>37</b>
4.1	CreateUserConfiguration Operation .....	37
4.2	DeleteUserConfiguration Operation .....	38
4.3	GetUserConfiguration Operation .....	39
4.4	UpdateUserConfiguration Operation .....	40
<b>5</b>	<b>Security .....</b>	<b>42</b>
5.1	Security Considerations for Implementers .....	42
5.2	Index of Security Parameters .....	42
<b>6</b>	<b>Appendix A: Full WSDL .....</b>	<b>43</b>
<b>7</b>	<b>Appendix B: Full XML Schema .....</b>	<b>46</b>
7.1	Messages Schema .....	46
7.2	Types Schema .....	47
<b>8</b>	<b>Appendix C: Product Behavior .....</b>	<b>50</b>

<b>9</b>	<b>Change Tracking.....</b>	<b>51</b>
<b>10</b>	<b>Index.....</b>	<b>52</b>

# 1 Introduction

The User Configuration Web Service Protocol is used to access user configuration objects. Clients use the SOAP Protocol, as described in [\[SOAP1.1\]](#), to contact the user configuration service.

Sections 1.5, 1.8, 1.9, 2, and 3 of this specification are normative. All other sections and examples in this specification are informative.

## 1.1 Glossary

This document uses the following terms:

**endpoint:** A communication port that is exposed by an application server for a specific shared service and to which messages can be addressed.

**folder associated information (FAI):** A collection of Message objects that are stored in a Folder object and are typically hidden from view by email applications. An FAI Message object is used to store a variety of settings and auxiliary data, including forms, views, calendar options, favorites, and category lists.

**Hypertext Transfer Protocol (HTTP):** An application-level protocol for distributed, collaborative, hypermedia information systems (text, graphic images, sound, video, and other multimedia files) on the World Wide Web.

**Hypertext Transfer Protocol Secure (HTTPS):** An extension of HTTP that securely encrypts and decrypts web page requests. In some older protocols, "Hypertext Transfer Protocol over Secure Sockets Layer" is still used (Secure Sockets Layer has been deprecated). For more information, see [\[SSL3\]](#) and [\[RFC5246\]](#).

**mailbox:** A message store that contains email, calendar items, and other Message objects for a single recipient.

**SOAP:** A lightweight protocol for exchanging structured information in a decentralized, distributed environment. **SOAP** uses **XML** technologies to define an extensible messaging framework, which provides a message construct that can be exchanged over a variety of underlying protocols. The framework has been designed to be independent of any particular programming model and other implementation-specific semantics. SOAP 1.2 supersedes SOAP 1.1. See [\[SOAP1.2-1/2003\]](#).

**SOAP action:** The HTTP request header field used to indicate the intent of the **SOAP** request, using a URI value. See [\[SOAP1.1\]](#) section 6.1.1 for more information.

**SOAP body:** A container for the payload data being delivered by a **SOAP message** to its recipient. See [\[SOAP1.2-1/2007\]](#) section 5.3 for more information.

**SOAP header:** A mechanism for implementing extensions to a **SOAP message** in a decentralized manner without prior agreement between the communicating parties. See [\[SOAP1.2-1/2007\]](#) section 5.2 for more information.

**SOAP message:** An **XML** document consisting of a mandatory SOAP envelope, an optional **SOAP header**, and a mandatory **SOAP body**. See [\[SOAP1.2-1/2007\]](#) section 5 for more information.

**Uniform Resource Locator (URL):** A string of characters in a standardized format that identifies a document or resource on the World Wide Web. The format is as specified in [\[RFC1738\]](#).

**web server:** A server computer that hosts websites and responds to requests from applications.

**Web Services Description Language (WSDL):** An XML format for describing network services as a set of endpoints that operate on messages that contain either document-oriented or

procedure-oriented information. The operations and messages are described abstractly and are bound to a concrete network protocol and message format in order to define an endpoint. Related concrete endpoints are combined into abstract endpoints, which describe a network service. WSDL is extensible, which allows the description of endpoints and their messages regardless of the message formats or network protocols that are used.

**WSDL message:** An abstract, typed definition of the data that is communicated during a WSDL operation [\[WSDL\]](#). Also, an element that describes the data being exchanged between web service providers and clients.

**WSDL port type:** A named set of logically-related, abstract **Web Services Description Language (WSDL)** operations and messages.

**XML:** The Extensible Markup Language, as described in [\[XML1.0\]](#).

**XML namespace:** A collection of names that is used to identify elements, types, and attributes in XML documents identified in a URI reference [\[RFC3986\]](#). A combination of XML namespace and local name allows XML documents to use elements, types, and attributes that have the same names but come from different sources. For more information, see [\[XMLNS-2ED\]](#).

**XML namespace prefix:** An abbreviated form of an **XML namespace**, as described in [\[XML\]](#).

**XML schema:** A description of a type of XML document that is typically expressed in terms of constraints on the structure and content of documents of that type, in addition to the basic syntax constraints that are imposed by **XML** itself. An XML schema provides a view of a document type at a relatively high level of abstraction.

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

## 1.2 References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the [Errata](#).

### 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](mailto:dochelp@microsoft.com). We will assist you in finding the relevant information.

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

[MS-OXWSCORE] Microsoft Corporation, "[Core Items Web Service Protocol](#)".

[MS-OXWSFOLD] Microsoft Corporation, "[Folders and Folder Permissions Web Service Protocol](#)".

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

[RFC2616] Fielding, R., Gettys, J., Mogul, J., et al., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999, <https://www.rfc-editor.org/info/rfc2616>

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000, <http://www.rfc-editor.org/rfc/rfc2818.txt>



[RFC3066] Alvestrand, H., "Tags for the Identification of Languages", BCP 47, RFC 3066, January 2001, <http://www.ietf.org/rfc/rfc3066.txt>

[SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", W3C Note, May 2000, <https://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, <https://www.w3.org/TR/2001/NOTE-wsdl-20010315>

[WSIBASIC] Ballinger, K., Ehnebuske, D., Gudgin, M., et al., Eds., "Basic Profile Version 1.0", Final Material, April 2004, <http://www.ws-i.org/Profiles/BasicProfile-1.0-2004-04-16.html>

[XMLNS] Bray, T., Hollander, D., Layman, A., et al., Eds., "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation, December 2009, <https://www.w3.org/TR/2009/REC-xml-names-20091208/>

[XMLSCHEMA1/2] Thompson, H., Beech, D., Maloney, M., and Mendelsohn, N., Eds., "XML Schema Part 1: Structures Second Edition", W3C Recommendation, October 2004, <https://www.w3.org/TR/2004/REC-xmlschema-1-20041028/>

[XMLSCHEMA2/2] Biron, P., and Malhotra, A., Eds., "XML Schema Part 2: Datatypes Second Edition", W3C Recommendation, October 2004, <https://www.w3.org/TR/2004/REC-xmlschema-2-20041028/>

### 1.2.2 Informative References

[MS-OXDSCLI] Microsoft Corporation, "[Autodiscover Publishing and Lookup Protocol](#)".

[MS-OXPROTO] Microsoft Corporation, "[Exchange Server Protocols System Overview](#)".

[MS-OXWSADISC] Microsoft Corporation, "[Autodiscover Publishing and Lookup SOAP-Based Web Service Protocol](#)".

## 1.3 Overview

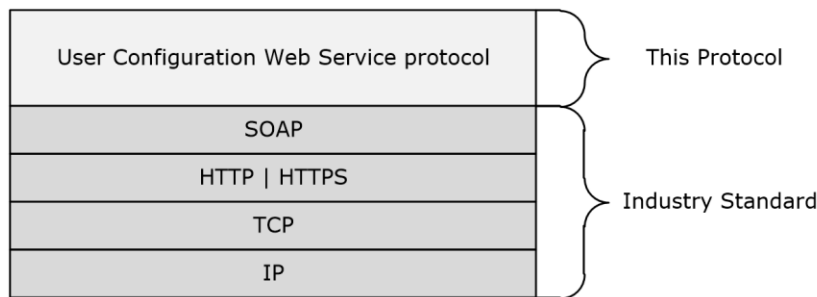
The User Configuration Web Service Protocol specifies the messages that are used to create, delete, get, and update user configuration objects, also known as **folder associated information (FAI)**.

## 1.4 Relationship to Other Protocols

A client that implements this protocol can use the Autodiscover Publishing and Lookup SOAP-Based Web Service Protocol, as described in [\[MS-OXWSADISC\]](#), or the Autodiscover Publishing and Lookup Protocol, as described in [\[MS-OXDSCLI\]](#), to identify the target **endpoint** to use for each operation.

This protocol uses **SOAP**, as described in [\[SOAP1.1\]](#), to specify the structure information exchanged between the client and the server. This protocol uses the **XML schema**, as described in [\[XMLSCHEMA1/2\]](#) and [\[XMLSCHEMA2/2\]](#), to describe the message content sent to and from the server.

This protocol uses SOAP over **HTTP**, as described in [\[RFC2616\]](#), and SOAP over **HTTPS**, as described in [\[RFC2818\]](#), as shown in the following layering diagram.



**Figure 2: This protocol in relation to other protocols**

For conceptual background information and overviews of the relationships and interactions between this and other protocols, see [\[MS-OXPROTO\]](#).

## 1.5 Prerequisites/Preconditions

The **endpoint URL** that is returned by either the Autodiscover Publishing Lookup SOAP-Based Web Service Protocol, as described in [\[MS-OXWSADISC\]](#), or the Autodiscover Publishing and Lookup Protocol, as described in [\[MS-OXDSCLI\]](#), is required to form the **HTTP** request to the **web server** that hosts this protocol. The operations that this protocol defines cannot be accessed unless the correct endpoint is identified in the HTTP web requests that target this protocol.

## 1.6 Applicability Statement

The protocol specified in this document is applicable to client applications that create, delete, get, or update **folder associated information (FAI)**. This Web service protocol is applicable to SOAP-based clients, as described in [\[SOAP1.1\]](#).

## 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. The **WSDL** version of the request is identified by using the **t:RequestServerVersion** element, as described in [\[MS-OXWSCDATA\]](#) section 2.2.3.9, and the version of the server responding to the request is identified by using the **t:ServerVersionInfo** element, as described in [\[MS-OXWSCDATA\]](#) section 2.2.3.10.
- **Security and Authentication Methods:** This protocol relies on the **web server** that is hosting it to perform authentication.
- **Capability Negotiation:** This protocol does not support version negotiation.

## 1.8 Vendor-Extensible Fields

This protocol consists of extensible fields. The applicable property types for extended fields are specified in section [3.1.4.3.4.1](#).

## 1.9 Standards Assignments

None.

## 2 Messages

In the following sections, the schema definition might differ from the processing rules imposed by the protocol. The **WSDL** in this specification provides a base description of the protocol. The schema in this specification provides a base description of the message syntax. The text that specifies the WSDL and schema might specify restrictions that reflect actual protocol behavior. For example, the schema definition might allow for an element to be **empty**, **null**, or **not present** but the behavior of the protocol as specified restricts the same elements to being **non-empty**, **not null**, or **present**.

### 2.1 Transport

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

This protocol MUST support **SOAP** over **HTTP**, as specified in [\[RFC2616\]](#). The protocol SHOULD use secure communications by means of **HTTPS**, as defined in [\[RFC2818\]](#).

### 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 [\[XMLSCHEMA1/2\]](#) and [\[XMLSCHEMA2/2\]](#), and **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** for each XML namespace that is used, the choice of any particular XML namespace prefix is implementation-specific and not significant for interoperability.

Prefix	Namespace URI	Reference
soap	<a href="http://schemas.xmlsoap.org/wsdl/soap/">http://schemas.xmlsoap.org/wsdl/soap/</a>	<a href="#">[SOAP1.1]</a>
m	<a href="http://schemas.microsoft.com/exchange/services/2006/messages">http://schemas.microsoft.com/exchange/services/2006/messages</a>	
t	<a href="http://schemas.microsoft.com/exchange/services/2006/types">http://schemas.microsoft.com/exchange/services/2006/types</a>	
xs	<a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a>	<a href="#">[XMLSCHEMA1/2]</a>
wsdl	<a href="http://schemas.xmlsoap.org/wsdl/">http://schemas.xmlsoap.org/wsdl/</a>	<a href="#">[WSDL]</a>
tns	<a href="http://schemas.microsoft.com/exchange/services/2006/messages">http://schemas.microsoft.com/exchange/services/2006/messages</a>	
wsi	<a href="http://ws-i.org/schemas/conformanceClaim/">http://ws-i.org/schemas/conformanceClaim/</a>	<a href="#">[WSIBASIC]</a>

#### 2.2.2 Messages

This specification does not define any common **WSDL message** definitions.

#### 2.2.3 Elements

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

## 2.2.4 Complex Types

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

Complex type name	Description
<b>UserConfigurationDictionaryEntryType</b>	Specifies the contents of a single dictionary entry property.
<b>UserConfigurationDictionaryObjectType</b>	Specifies a dictionary object.
<b>UserConfigurationDictionaryType</b>	Specifies a set of dictionary properties.
<b>UserConfigurationNameType</b>	Specifies the name and the associated folder for a user configuration object.
<b>UserConfigurationType</b>	Specifies a single user configuration object.

### 2.2.4.1 t:UserConfigurationDictionaryEntryType Complex Type

The **UserConfigurationDictionaryEntryType** complex type specifies the contents of a single dictionary entry property.

```
<xs:complexType name="UserConfigurationDictionaryEntryType">
  <xs:sequence>
    <xs:element name="DictionaryKey"
      type="t:UserConfigurationDictionaryObjectType"
    />
    <xs:element name="DictionaryValue"
      type="t:UserConfigurationDictionaryObjectType"
      nillable="true"
    />
  </xs:sequence>
</xs:complexType>
```

The following table lists and describes the child elements of the **UserConfigurationDictionaryEntryType** complex type.

Element name	Type	Description
<b>DictionaryKey</b>	<b>t:UserConfigurationDictionaryObjectType</b> (section <a href="#">2.2.4.2</a> )	Specifies the dictionary key for a dictionary property.
<b>DictionaryValue</b>	<b>t:UserConfigurationDictionaryObjectType</b> (section <a href="#">2.2.4.2</a> )	Specifies the dictionary value for a dictionary property.

### 2.2.4.2 t:UserConfigurationDictionaryObjectType Complex Type

The **UserConfigurationDictionaryObjectType** complex type specifies a dictionary object.

```
<xs:complexType name="UserConfigurationDictionaryObjectType">
  <xs:sequence>
    <xs:element name="Type"

```

```

        type="t:UserConfigurationDictionaryObjectTypesType"
      />
    <xs:element name="Value"
      type="xs:string"
      maxOccurs="unbounded"
    />
  </xs:sequence>
</xs:complexType>

```

The following table lists the child elements of the **UserConfigurationDictionaryObjectType** complex type.

Element name	Type	Description
<b>Type</b>	<b>t:UserConfigurationDictionaryObjectTypesType</b> (section <a href="#">2.2.5.1</a> )	Specifies the dictionary object type.
<b>Value</b>	<b>xs:string</b> ( <a href="#">XMLSCHEMA2/2</a> )	Specifies the dictionary object value as a string.

### 2.2.4.3 t:UserConfigurationDictionaryType Complex Type

The **UserConfigurationDictionaryType** complex type specifies a set of dictionary properties.

```

<xs:complexType name="UserConfigurationDictionaryType">
  <xs:sequence>
    <xs:element name="DictionaryEntry"
      type="t:UserConfigurationDictionaryEntryType"
      minOccurs="0"
      maxOccurs="unbounded"
    />
  </xs:sequence>
</xs:complexType>

```

The following table lists and describes the child elements of the **UserConfigurationDictionaryType** complex type.

Element name	Type	Description
<b>DictionaryEntry</b>	<b>t:UserConfigurationDictionaryEntryType</b> (section <a href="#">2.2.4.1</a> )	Specifies the properties of the dictionary entry.

### 2.2.4.4 t:UserConfigurationNameType Complex Type

The **UserConfigurationNameType** complex type specifies the name and the associated folder for a user configuration object. This type extends the **TargetFolderIdType** complex type, as specified in [\[MS-OXWSFOLD\]](#) section 2.2.4.16.

```

<xs:complexType name="UserConfigurationNameType">
  <xs:complexContent>
    <xs:extension

```

```

        base="t:TargetFolderIdType"
    >
        <xs:attribute name="Name"
            type="t:NonEmptyStringType"
            use="required"
        />
    </xs:extension>
</xs:complexContent>
</xs:complexType>

```

The following table lists the attributes of the **UserConfigurationNameType** complex type.

Attribute name	Type	Description
<b>Name</b>	<b>t:NonEmptyStringType</b> ( <a href="#">[MS-OXWSCDATA]</a> section 2.2.5.20)	Specifies the name of a user configuration object. This attribute <b>MUST</b> be present.

#### 2.2.4.5 t:UserConfigurationType Complex Type

The **UserConfigurationType** complex type specifies a single user configuration object.

```

<xs:complexType name="UserConfigurationType">
    <xs:sequence>
        <xs:element name="UserConfigurationName"
            type="t:UserConfigurationNameType"
        />
        <xs:element name="ItemId"
            type="t:ItemIdType"
            minOccurs="0"
        />
        <xs:element name="Dictionary"
            type="t:UserConfigurationDictionaryType"
            minOccurs="0"
        />
        <xs:element name="XmlData"
            type="xs:base64Binary"
            minOccurs="0"
        />
        <xs:element name="BinaryData"
            type="xs:base64Binary"
            minOccurs="0"
        />
    </xs:sequence>
</xs:complexType>

```

The following table lists the child elements of the **UserConfigurationType** complex type.

Element name	Type	Description
<b>UserConfigurationName</b>	<b>t:UserConfigurationNameType</b> (section <a href="#">2.2.4.4</a> )	Specifies the name and the associated folder for a user configuration object. The user configuration object name <b>MUST</b> be present.
<b>ItemId</b>	<b>t:ItemIdType</b> ( <a href="#">[MS-OXWSCORE]</a> section 2.2.4.25)	Specifies the user configuration object item identifier.

Element name	Type	Description
<b>Dictionary</b>	<b>t:UserConfigurationDictionaryType</b> (section <a href="#">2.2.4.3</a> )	Specifies a set of dictionary property entries.
<b>XmlData</b>	<b>xs:base64Binary</b> ( <a href="#">[XMLSCHEMA2/2]</a> )	Specifies XML data property content.
<b>BinaryData</b>	<b>xs:base64Binary</b> ( <a href="#">[XMLSCHEMA2/2]</a> )	Specifies binary data property content.

## 2.2.5 Simple Types

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

Simple type name	Description
<b>UserConfigurationDictionaryObjectTypesType</b>	Specifies the data type for a user configuration object dictionary property.

### 2.2.5.1 t:UserConfigurationDictionaryObjectTypesType Simple Type

The **UserConfigurationDictionaryObjectTypesType** simple type specifies the data type for a user configuration object dictionary property.

```
<xs:simpleType name="UserConfigurationDictionaryObjectTypesType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="DateTime"
    />
    <xs:enumeration
      value="Boolean"
    />
    <xs:enumeration
      value="Byte"
    />
    <xs:enumeration
      value="String"
    />
    <xs:enumeration
      value="Integer32"
    />
    <xs:enumeration
      value="UnsignedInteger32"
    />
    <xs:enumeration
      value="Integer64"
    />
    <xs:enumeration
      value="UnsignedInteger64"
    />
    <xs:enumeration
      value="StringArray"
    />
    <xs:enumeration
      value="ByteArray"
    />
  </xs:restriction>
</xs:simpleType>
```

```
</xs:restriction>  
</xs:simpleType>
```

The following table lists and describes the values that are defined by the **UserConfigurationDictionaryObjectTypesType** simple type.

Value name	Description
<b>DateTime</b>	Specifies a DateTime data type for a dictionary entry.
<b>Boolean</b>	Specifies a Boolean data type for a dictionary entry.
<b>Byte</b>	Specifies a byte data type for a dictionary entry.
<b>String</b>	Specifies a string data type for a dictionary entry.
<b>Integer32</b>	Specifies a signed 32-bit integer data type for a dictionary entry.
<b>UnsignedInteger32</b>	Specifies an unsigned 32-bit integer data type for a dictionary entry.
<b>Integer64</b>	Specifies a signed 64-bit integer data type for a dictionary entry.
<b>UnsignedInteger64</b>	Specifies an unsigned 64-bit integer data type for a dictionary entry.
<b>StringArray</b>	Specifies a string array data type for a dictionary entry.
<b>ByteArray</b>	Specifies a byte array data type for a dictionary entry.

## 2.2.6 Attributes

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

## 2.2.7 Groups

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

## 2.2.8 Attribute Groups

This specification does not define any common **XML schema** attribute group 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 User Configuration Web Service Protocol defines a single **WSDL port type** with four operations. These operations allow users to create, delete, retrieve, and update users configuration objects.

#### 3.1.1 Abstract Data Model

The User Configuration Web Service Protocol is a stateless protocol. The server maintains the user configuration object data.

The client is not required to maintain the state of the user configuration objects on the server and can retrieve the current set of configuration objects at any time. If more than one client accesses user configuration objects at the same time, there is no requirement that the server lock the existing set of permissions for changes.

#### 3.1.2 Timers

None.

#### 3.1.3 Initialization

None.

#### 3.1.4 Message Processing Events and Sequencing Rules

The following table summarizes the **WSDL** operations as defined by this specification.

Operation	Description
<b>CreateUserConfiguration</b> (section <a href="#">3.1.4.1</a> )	Specifies the <b>WSDL messages</b> for creating user configuration objects.
<b>DeleteUserConfiguration</b> (section <a href="#">3.1.4.2</a> )	Specifies the WSDL messages for deleting user configuration objects.
<b>GetUserConfiguration</b> (section <a href="#">3.1.4.3</a> )	Specifies the WSDL messages for getting user configuration objects.
<b>UpdateUserConfiguration</b> (section <a href="#">3.1.4.4</a> )	Specifies the WSDL messages for updating user configuration objects.

##### 3.1.4.1 CreateUserConfiguration

The **CreateUserConfiguration** operation specifies the **WSDL messages** for creating user configuration objects.

The following is the **WSDL port type** definition of this operation.

```

<wsdl:operation name="CreateUserConfiguration">
  <wsdl:input message="tns:CreateUserConfigurationSoapIn"/>
  <wsdl:output message="tns:CreateUserConfigurationSoapOut"/>
</wsdl:operation>

```

The following is the **WSDL** binding specification for this operation.

```

<wsdl:operation name="CreateUserConfiguration">
  <soap:operation
    soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/CreateUserConfiguration" />
  <wsdl:input>
    <soap:header message="tns:CreateUserConfigurationSoapIn" part="Impersonation"
      use="literal"/>
    <soap:header message="tns:CreateUserConfigurationSoapIn" part="MailboxCulture"
      use="literal"/>
    <soap:header message="tns:CreateUserConfigurationSoapIn" part="RequestVersion"
      use="literal"/>
    <soap:body parts="request" use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap:body parts="CreateUserConfigurationResult" use="literal" />
    <soap:header message="tns:CreateUserConfigurationSoapOut" part="ServerVersion"
      use="literal"/>
  </wsdl:output>
</wsdl:operation>

```

The protocol client sends a **CreateUserConfigurationSoapIn** request WSDL message, as specified in section [3.1.4.1.1.1](#), and the protocol server responds with a **CreateUserConfigurationSoapOut** response message, as defined in section [3.1.4.1.1.2](#).

### 3.1.4.1.1 Messages

The following table lists and describes the **WSDL message** definitions that are specific to the **CreateUserConfiguration** operation.

Message	Description
<b>CreateUserConfigurationSoapIn</b>	Specifies the <b>SOAP message</b> that creates a user configuration object.
<b>CreateUserConfigurationSoapOut</b>	Specifies the SOAP message that is returned by the server in response to a <b>CreateUserConfiguration</b> operation request.

#### 3.1.4.1.1.1 CreateUserConfigurationSoapIn

The **CreateUserConfigurationSoapIn** **WSDL message** specifies the **CreateUserConfiguration** operation request to create a user configuration object.

```

<wsdl:message name="CreateUserConfigurationSoapIn">
  <wsdl:part name="request" element="tns:CreateUserConfiguration" />
  <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
  <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
  <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
</wsdl:message>

```

The **CreateUserConfigurationSoapIn** WSDL message is the input message for the **SOAP action** <http://schemas.microsoft.com/exchange/services/2006/messages/CreateUserConfiguration>.

The parts of the **CreateUserConfigurationSoapIn** WSDL message are listed and described in the following table.

Part name	Element/type	Description
<b>request</b>	<b>tns:CreateUserConfiguration</b> (section <a href="#">3.1.4.1.2.1</a> )	Specifies the <b>SOAP body</b> of the request to create a user configuration object. This part is required.
<b>Impersonation</b>	<b>t:ExchangeImpersonation</b> ( <a href="#">[MS-OXWSCDATA]</a> section 2.2.3.3)	Specifies a <b>SOAP header</b> that identifies the user whom the client application is impersonating. This part is optional.
<b>MailboxCulture</b>	<b>t:MailboxCulture</b> ( <a href="#">[MS-OXWSCDATA]</a> section 2.2.3.6)	Specifies a SOAP header that identifies the culture to use for accessing the <b>mailbox</b> . The cultures are defined in <a href="#">[RFC3066]</a> . This part is optional.
<b>RequestVersion</b>	<b>t:RequestServerVersion</b> ( <a href="#">[MS-OXWSCDATA]</a> section 2.2.3.9)	Specifies a SOAP header that identifies the schema version for the <b>CreateUserConfiguration</b> operation request.

### 3.1.4.1.1.2 CreateUserConfigurationSoapOut

The **CreateUserConfigurationSoapOut** **WSDL message** specifies the server response to a **CreateUserConfiguration** operation request to create a user configuration object.

```
<wsdl:message name="CreateUserConfigurationSoapOut">
  <wsdl:part name="CreateUserConfigurationResult"
    element="tns:CreateUserConfigurationResponse" />
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
```

The **CreateUserConfigurationSoapOut** WSDL message is the output message for the **SOAP action** <http://schemas.microsoft.com/exchange/services/2006/messages/CreateUserConfiguration>.

The parts of the **CreateUserConfigurationSoapOut** WSDL message are listed and described in the following table.

Part name	Element/type	Description
<b>CreateUserConfigurationResult</b>	<b>tns:CreateUserConfigurationResponse</b> (section <a href="#">3.1.4.1.2.2</a> )	Specifies the <b>SOAP body</b> of the response to a <b>CreateUserConfiguration</b> operation request.
<b>ServerVersion</b>	<b>t:ServerVersionInfo</b> ( <a href="#">[MS-OXWSCDATA]</a> section 2.2.3.10)	Specifies the <b>SOAP header</b> that identifies the server version for the response to a <b>CreateUserConfiguration</b> request.

A successful **CreateUserConfiguration** operation request returns a **CreateUserConfigurationResponse** element, as specified in section 3.1.4.1.2.2, with the **ResponseClass** attribute of the **CreateUserConfigurationResponseMessage** element set to

"Success". The **ResponseCode** element of the **CreateUserConfigurationResponseMessage** element is set to "NoError". The **CreateUserConfigurationResponseMessage** element is a child element of the **ArrayOfResponseMessagesType** complex type, as specified in [MS-OXWSCDATA] section 2.2.4.12.

If the **CreateUserConfiguration** operation request is not successful, it returns a **CreateUserConfigurationResponse** element with the **ResponseClass** attribute of the **CreateUserConfigurationResponseMessage** element set to "Error". The **ResponseCode** element of the **CreateUserConfigurationResponseMessage** element is set to one of the common errors defined in [MS-OXWSCDATA] section 2.2.5.24.

### 3.1.4.1.2 Elements

The following table summarizes the **XML schema** element definitions that are specific to the **CreateUserConfiguration** operation.

Element	Description
<b>CreateUserConfiguration</b>	Specifies a request to create a user configuration object.
<b>CreateUserConfigurationResponse</b>	Specifies a response to a <b>CreateUserConfiguration</b> operation request.

#### 3.1.4.1.2.1 CreateUserConfiguration Element

The **CreateUserConfiguration** element specifies a request to create a user configuration object.

```
<xs:element name="CreateUserConfiguration"
  type="m:CreateUserConfigurationType"
/>
```

#### 3.1.4.1.2.2 CreateUserConfigurationResponse Element

The **CreateUserConfigurationResponse** element specifies a response to a **CreateUserConfiguration** operation request (section [3.1.4.1](#)).

```
<xs:element name="CreateUserConfigurationResponse"
  type="m:CreateUserConfigurationResponseType"
/>
```

#### 3.1.4.1.3 Complex Types

The following table summarizes the **XML schema** complex type definitions that are specific to the **CreateUserConfiguration** operation.

Complex type	Description
<b>CreateUserConfigurationResponseType</b>	Specifies a response to a <b>CreateUserConfiguration</b> operation request.
<b>CreateUserConfigurationType</b>	Specifies a <b>CreateUserConfiguration</b> operation request.

### 3.1.4.1.3.1 m:CreateUserConfigurationResponseType Complex Type

The **CreateUserConfigurationResponseType** complex type specifies the response to a **CreateUserConfiguration** operation request (section 3.1.4.1). The **CreateUserConfigurationResponseType** complex type extends the **BaseResponseMessageType** complex type, as specified in [MS-OXWSCDATA] section 2.2.4.18.

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

### 3.1.4.1.3.2 m:CreateUserConfigurationType Complex Type

The **CreateUserConfigurationType** complex type specifies a **CreateUserConfiguration** operation request (section 3.1.4.1). The **CreateUserConfigurationType** complex type extends the **BaseRequestType** complex type, as specified in [MS-OXWSCDATA] section 2.2.4.17.

```
<xs:complexType name="CreateUserConfigurationType">
  <xs:complexContent>
    <xs:extension
      base="m:BaseRequestType"
    >
      <xs:sequence>
        <xs:element name="UserConfiguration"
          type="t:UserConfigurationType"
        />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The following table summarizes the child element of the **CreateUserConfigurationType** complex type.

Element	Type	Description
<b>UserConfiguration</b>	t: <b>UserConfigurationType</b> (section 2.2.4.5)	Specifies the user configuration object to be created.

### 3.1.4.1.4 Simple Types

None.

### 3.1.4.1.5 Attributes

None.

### 3.1.4.1.6 Groups

None

### 3.1.4.1.7 Attribute Groups

None.

### 3.1.4.2 DeleteUserConfiguration

The **DeleteUserConfiguration** operation specifies the **WSDL messages** for deleting user configuration objects.

The following is the **WSDL port type** definition of this operation.

```
<wsdl:operation name="DeleteUserConfiguration">
  <wsdl:input message="tns:DeleteUserConfigurationSoapIn"/>
  <wsdl:output message="tns:DeleteUserConfigurationSoapOut"/>
</wsdl:operation>
```

The following is the **WSDL binding** specification for this operation.

```
<wsdl:operation name="DeleteUserConfiguration">
  <soap:operation
    soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/DeleteUserConfiguration" />
  <wsdl:input>
    <soap:header message="tns:DeleteUserConfigurationSoapIn" part="Impersonation"
      use="literal"/>
    <soap:header message="tns:DeleteUserConfigurationSoapIn" part="MailboxCulture"
      use="literal"/>
    <soap:header message="tns:DeleteUserConfigurationSoapIn" part="RequestVersion"
      use="literal"/>
    <soap:body parts="request" use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap:body parts="DeleteUserConfigurationResult" use="literal" />
    <soap:header message="tns:DeleteUserConfigurationSoapOut" part="ServerVersion"
      use="literal"/>
  </wsdl:output>
</wsdl:operation>
```

The protocol client sends a DeleteUserConfigurationSoapIn request WSDL message, as specified in section [3.1.4.2.1.1](#), and the protocol server responds with a DeleteUserConfigurationSoapOut response message, as defined in section [3.1.4.2.1.2](#).

#### 3.1.4.2.1 Messages

The following table lists and describes the **WSDL message** definitions that are specific to the **DeleteUserConfiguration** operation.

Message	Description
<b>DeleteUserConfigurationSoapIn</b>	Specifies the <b>SOAP message</b> that deletes a user configuration object.
<b>DeleteUserConfigurationSoapOut</b>	Specifies the SOAP message that is returned by the server in response to a <b>DeleteUserConfiguration</b> operation request.

#### 3.1.4.2.1.1 DeleteUserConfigurationSoapIn

The **DeleteUserConfigurationSoapIn WSDL message** specifies a **DeleteUserConfiguration** operation request to delete a user configuration object.

```
<wsdl:message name="DeleteUserConfigurationSoapIn">
  <wsdl:part name="request" element="tns:DeleteUserConfiguration" />
  <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
  <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
  <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
</wsdl:message>
```

The **DeleteUserConfigurationSoapIn** WSDL message is the input message for the **SOAP action** <http://schemas.microsoft.com/exchange/services/2006/messages/DeleteUserConfiguration>.

The parts of the **DeleteUserConfigurationSoapIn** WSDL message are listed and described in the following table.

Part name	Element/type	Description
<b>request</b>	<b>tns:DeleteUserConfiguration</b> (section <a href="#">3.1.4.2.2.1</a> )	Specifies the <b>SOAP body</b> of a request to delete a user configuration object. This part is required.
<b>Impersonation</b>	<b>t:ExchangeImpersonation</b> ([MS-OXWSCDATA] section 2.2.3.3)	Specifies a <b>SOAP header</b> that identifies the user whom the client application is impersonating. This part is optional.
<b>MailboxCulture</b>	<b>t:MailboxCulture</b> ([MS-OXWSCDATA] section 2.2.3.6)	Specifies a SOAP header that identifies the culture to be used for accessing the <b>mailbox</b> . The cultures are defined in <a href="#">[RFC3066]</a> . This part is optional.
<b>RequestVersion</b>	<b>t:RequestServerVersion</b> ([MS-OXWSCDATA] section 2.2.3.9)	Specifies a SOAP header that identifies the schema version for the <b>DeleteUserConfiguration</b> operation request (section <a href="#">3.1.4.2</a> ).

### 3.1.4.2.1.2 DeleteUserConfigurationSoapOut

The **DeleteUserConfigurationSoapOut WSDL message** specifies the server response to a **DeleteUserConfiguration** operation request to delete a user configuration object.

```
<wsdl:message name="DeleteUserConfigurationSoapOut">
  <wsdl:part name="DeleteUserConfigurationResult"
    element="tns:DeleteUserConfigurationResponse" />
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
```

The **DeleteUserConfigurationSoapOut** WSDL message is the output message for the **SOAP action** <http://schemas.microsoft.com/exchange/services/2006/messages/DeleteUserConfiguration>.

The parts of the **DeleteUserConfigurationSoapOut** WSDL message are listed and described in the following table.

Part name	Element/type	Description
<b>DeleteUserConfigurationResult</b>	<b>tns:DeleteUserConfigurationResponse</b> (section <a href="#">3.1.4.2.2.2</a> )	Specifies the <b>SOAP body</b> of the response to a <b>DeleteUserConfiguration</b> operation request.

Part name	Element/type	Description
<b>ServerVersion</b>	<b>t:ServerVersionInfo</b> ( <a href="#">[MS-OXWSCDATA]</a> section 2.2.3.10)	Specifies the <b>SOAP header</b> that identifies the server version for the response.

A successful **DeleteUserConfiguration** operation request returns a **DeleteUserConfigurationResponse** element, as specified in section 3.1.4.2.2.2, with the **ResponseClass** attribute of the **DeleteUserConfigurationResponseMessage** element set to "Success". The **ResponseCode** element of the **DeleteUserConfigurationResponseMessage** element is set to "NoError". The **DeleteUserConfigurationResponseMessage** element is a child element of the **ArrayOfResponseMessagesType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.4.12.

If the **DeleteUserConfiguration** operation request is not successful, it returns a **DeleteUserConfigurationResponse** element with the **ResponseClass** attribute of the **DeleteUserConfigurationResponseMessage** element set to "Error". The **ResponseCode** element of the **DeleteUserConfigurationResponseMessage** element is set to one of the common errors defined in [\[MS-OXWSCDATA\]](#) section 2.2.5.24.

### 3.1.4.2.2 Elements

The following table summarizes the **XML schema** element definitions that are specific to the **DeleteUserConfiguration** operation.

Element	Description
<b>DeleteUserConfiguration</b>	Specifies a request to delete a user configuration object.
<b>DeleteUserConfigurationResponse</b>	Specifies the response to a <b>DeleteUserConfiguration</b> operation request.

#### 3.1.4.2.2.1 DeleteUserConfiguration Element

The **DeleteUserConfiguration** element specifies a request to delete a user configuration object.

```
<xs:element name="DeleteUserConfiguration"
  type="m:DeleteUserConfigurationType"
/>
```

#### 3.1.4.2.2.2 DeleteUserConfigurationResponse Element

The **DeleteUserConfigurationResponse** element specifies the response to a **DeleteUserConfiguration** operation request (section [3.1.4.2](#)).

```
<xs:element name="DeleteUserConfigurationResponse"
  type="m:DeleteUserConfigurationResponseType"
/>
```

### 3.1.4.2.3 Complex Types

The following table summarizes the **XML schema** complex type definitions that are specific to the **DeleteUserConfiguration** operation.



Complex type	Description
<b>DeleteUserConfigurationResponseType</b>	Specifies the response to a <b>DeleteUserConfiguration</b> operation request.
<b>DeleteUserConfigurationType</b>	Specifies a <b>DeleteUserConfiguration</b> operation request.

### 3.1.4.2.3.1 m:DeleteUserConfigurationResponseType Complex Type

The **DeleteUserConfigurationResponseType** complex type specifies the response for a **DeleteUserConfiguration** operation request (section 3.1.4.2). The **DeleteUserConfigurationResponseType** complex type extends the **BaseResponseMessageType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.4.18.

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

### 3.1.4.2.3.2 m:DeleteUserConfigurationType Complex Type

The **DeleteUserConfigurationType** complex type specifies a **DeleteUserConfiguration** operation request (section 3.1.4.2). The **DeleteUserConfigurationType** complex type extends the **BaseRequestType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.4.17.

```
<xs:complexType name="DeleteUserConfigurationType">
  <xs:complexContent>
    <xs:extension
      base="m:BaseRequestType"
    >
      <xs:sequence>
        <xs:element name="UserConfigurationName"
          type="t:UserConfigurationNameType"
        />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The following table summarizes the child element of the **DeleteUserConfigurationType** complex type.

Element	Type	Description
<b>UserConfigurationName</b>	<b>t:UserConfigurationNameType</b> (section 2.2.4.4)	Specifies the name and the associated folder for the user configuration object to be deleted. The user configuration object name MUST be present.

### 3.1.4.2.4 Simple Types

None.

#### 3.1.4.2.5 Attributes

None

#### 3.1.4.2.6 Groups

None.

#### 3.1.4.2.7 Attribute Groups

None.

### 3.1.4.3 GetUserConfiguration

The **GetUserConfiguration** operation specifies the **WSDL messages** that are used to get user configuration objects.

The following is the **WSDL port type** definition of this operation.

```
<wsdl:operation name="GetUserConfiguration">
  <wsdl:input message="tns:GetUserConfigurationSoapIn"/>
  <wsdl:output message="tns:GetUserConfigurationSoapOut"/>
</wsdl:operation>
```

The following is the **WSDL binding** specification for this operation.

```
<wsdl:operation name="GetUserConfiguration">
  <soap:operation
    soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/GetUserConfiguration"
  />
  <wsdl:input>
    <soap:header message="tns:GetUserConfigurationSoapIn" part="Impersonation"
      use="literal"/>
    <soap:header message="tns:GetUserConfigurationSoapIn" part="MailboxCulture"
      use="literal"/>
    <soap:header message="tns:GetUserConfigurationSoapIn" part="RequestVersion"
      use="literal"/>
    <soap:body parts="request" use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap:body parts="GetUserConfigurationResult" use="literal" />
    <soap:header message="tns:GetUserConfigurationSoapOut" part="ServerVersion"
      use="literal"/>
  </wsdl:output>
</wsdl:operation>
```

The protocol client sends a GetUserConfigurationSoapIn request WSDL message, as specified in section [3.1.4.3.1.1](#), and the protocol server responds with a GetUserConfigurationSoapOut response message, as defined in section [3.1.4.3.1.2](#).

#### 3.1.4.3.1 Messages

The following table lists and describes the **WSDL message** definitions that are specific to the **GetUserConfiguration** operation.

Message	Description
<b>GetUserConfigurationSoapIn</b>	Specifies the <b>SOAP message</b> that gets a user configuration object.
<b>GetUserConfigurationSoapOut</b>	Specifies the SOAP message that is returned by the server in response to the <b>GetUserConfiguration</b> operation request.

### 3.1.4.3.1.1 GetUserConfigurationSoapIn

The **GetUserConfigurationSoapIn WSDL message** specifies a **GetUserConfiguration** operation request to retrieve a user configuration object.

```
<wsdl:message name="GetUserConfigurationSoapIn">
  <wsdl:part name="request" element="tns:GetUserConfiguration" />
  <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
  <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
  <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
</wsdl:message>
```

The **GetUserConfigurationSoapIn WSDL message** is the input message for the **SOAP action** <http://schemas.microsoft.com/exchange/services/2006/messages/GetUserConfiguration>.

The parts of the **GetUserConfigurationSoapIn WSDL message** are listed and described in the following table.

Part name	Element/type	Description
<b>request</b>	<b>tns:GetUserConfiguration</b> (section <a href="#">3.1.4.3.2.1</a> )	Specifies the <b>SOAP body</b> of a request to get a user configuration object. This part is required.
<b>Impersonation</b>	<b>t:ExchangeImpersonation</b> ([MS-OXWSCDATA] section 2.2.3.3)	Specifies a <b>SOAP header</b> that identifies the user whom the client application is impersonating. This part is optional.
<b>MailboxCulture</b>	<b>t:MailboxCulture</b> ([MS-OXWSCDATA] section 2.2.3.6)	Specifies a SOAP header that identifies the culture to be used for accessing the <b>mailbox</b> . The cultures are defined in <a href="#">[RFC3066]</a> . This part is optional.
<b>RequestVersion</b>	<b>t:RequestServerVersion</b> ([MS-OXWSCDATA] section 2.2.3.9)	Specifies a SOAP header that identifies the schema version for the <b>GetUserConfiguration</b> operation request (section <a href="#">3.1.4.3</a> ).

### 3.1.4.3.1.2 GetUserConfigurationSoapOut

The **GetUserConfigurationSoapOut WSDL message** specifies the server response to a **GetUserConfiguration** operation request to get a user configuration object.

```
<wsdl:message name="GetUserConfigurationSoapOut">
  <wsdl:part name="GetUserConfigurationResult" element="tns:GetUserConfigurationResponse"/>
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
```

The parts of the **GetUserConfigurationSoapOut WSDL message** are listed and described in the following table.

Part name	Element/type	Description
<b>GetUserConfigurationResult</b>	<b>tns:GetUserConfigurationResponse</b> (section <a href="#">3.1.4.3.2.2</a> )	Specifies the <b>SOAP body</b> of the response to a <b>GetUserConfiguration</b> operation request.
<b>ServerVersion</b>	<b>t:ServerVersionInfo</b> ( <a href="#">[MS-OXWSCDATA]</a> section 2.2.3.10)	Specifies the <b>SOAP header</b> that identifies the server version for the response.

A successful **GetUserConfiguration** operation request returns a **GetUserConfigurationResponse** element, as specified in section 3.1.4.3.2.2, with the **ResponseClass** attribute of the **GetUserConfigurationResponseMessage** element set to "Success". The **ResponseCode** element of the **GetUserConfigurationResponseMessage** element is set to "NoError". The **GetUserConfigurationResponseMessage** element is a child element of the **ArrayOfResponseMessagesType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.4.12.

If the **GetUserConfiguration** operation request is not successful, it returns a **GetUserConfigurationResponse** element with the **ResponseClass** attribute of the **GetUserConfigurationResponseMessage** element set to "Error". The **ResponseCode** element of the **GetUserConfigurationResponseMessage** element is set to one of the common errors defined in [\[MS-OXWSCDATA\]](#) section 2.2.5.24.

### 3.1.4.3.2 Elements

The following table summarizes the **XML schema** element definitions that are specific to the **GetUserConfiguration** operation.

Element	Description
<b>GetUserConfiguration</b>	Specifies a request to get a user configuration object.
<b>GetUserConfigurationResponse</b>	Specifies the response to a <b>GetUserConfiguration</b> operation request.

#### 3.1.4.3.2.1 GetUserConfiguration Element

The **GetUserConfiguration** element specifies a request to get a user configuration object.

```
<xs:element name="GetUserConfiguration"
  type="m:GetUserConfigurationType"
/>
```

#### 3.1.4.3.2.2 GetUserConfigurationResponse Element

The **GetUserConfigurationResponse** element specifies the response to a **GetUserConfiguration** operation request (section [3.1.4.3](#)).

```
<xs:element name="GetUserConfigurationResponse"
  type="m:GetUserConfigurationResponseType"
/>
```

### 3.1.4.3.3 Complex Types

The following table summarizes the **XML schema** complex type definitions that are specific to the **GetUserConfiguration** operation.

Complex type	Description
<b>GetUserConfigurationResponseMessageType</b>	Specifies a user configuration object returned in a response.
<b>GetUserConfigurationResponseType</b>	Specifies a response to a <b>GetUserConfiguration</b> operation request.
<b>GetUserConfigurationType</b>	Specifies a <b>GetUserConfiguration</b> operation request.

#### 3.1.4.3.3.1 m:GetUserConfigurationResponseMessageType Complex Type

The **GetUserConfigurationResponseMessageType** complex type specifies a user configuration object that is returned in a response to a **GetUserConfiguration** operation request. The **GetUserConfigurationResponseMessageType** complex type extends the **ResponseMessageType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.4.65.

```
<xs:complexType name="GetUserConfigurationResponseMessageType">
  <xs:complexContent>
    <xs:extension
      base="m:ResponseMessageType"
    >
      <xs:sequence>
        <xs:element name="UserConfiguration"
          type="t:UserConfigurationType"
          minOccurs="0"
        />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The following table summarizes the child element of the **GetUserConfigurationResponseMessageType** complex type.

Element	Type	Description
<b>UserConfiguration</b>	<b>t:UserConfigurationType</b> (section <a href="#">2.2.4.5</a> )	Specifies a single user configuration object.

#### 3.1.4.3.3.2 m:GetUserConfigurationType Complex Type

The **GetUserConfigurationType** complex type specifies a **GetUserConfiguration** operation request (section [3.1.4.3](#)). The **GetUserConfigurationType** complex type extends the **BaseRequestType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.4.17.

```
<xs:complexType name="GetUserConfigurationType">
  <xs:complexContent>
    <xs:extension
      base="m:BaseRequestType"
    >
      <xs:sequence>
```

```

<xs:element name="UserConfigurationName"
  type="t:UserConfigurationNameType"
/>
<xs:element name="UserConfigurationProperties"
  type="t:UserConfigurationPropertyType"
/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

```

The following table summarizes the child elements of the GetUserConfigurationType complex type.

Element	Type	Description
<b>UserConfigurationName</b>	<b>t:UserConfigurationNameType</b> (section <a href="#">2.2.4.4</a> )	Specifies the name and the associated folder for a user configuration object. The user configuration object name <b>MUST</b> be present.
<b>UserConfigurationProperties</b>	<b>t:UserConfigurationPropertyType</b> (section <a href="#">3.1.4.3.4.1</a> )	Specifies the property types to get in a <b>GetUserConfiguration</b> operation (section 3.1.4.3).

### 3.1.4.3.4 Simple Types

The following table summarizes the **XML schema** complex type definition that is specific to the **GetUserConfiguration** operation.

Simple type name	Description
<b>UserConfigurationPropertyType</b>	Specifies the property types to get in a <b>GetUserConfiguration</b> operation, as specified in section <a href="#">3.1.4.3</a> .

#### 3.1.4.3.4.1 t:UserConfigurationPropertyType Simple Type

The **UserConfigurationPropertyType** simple type specifies the property types to get in a **GetUserConfiguration** operation, as specified in section [3.1.4.3](#).

```

<xs:simpleType name="UserConfigurationPropertyType">
  <xs:list>
    <xs:simpleType>
      <xs:restriction
        base="xs:string"
      >
        <xs:enumeration
          value="Id"
        />
        <xs:enumeration
          value="Dictionary"
        />
        <xs:enumeration
          value="XmlData"
        />
        <xs:enumeration

```

```

        value="BinaryData"
      />
    <xs:enumeration
      value="All"
    />
  </xs:restriction>
</xs:simpleType>
</xs:list>
</xs:simpleType>

```

The following table lists and describes the values that are defined by the **UserConfigurationPropertyType** simple type.

Value name	Description
<b>Id</b>	Specifies the identifier property.
<b>Dictionary</b>	Specifies dictionary property types.
<b>XmlData</b>	Specifies <b>XML</b> data property types.
<b>BinaryData</b>	Specifies binary data property types.
<b>All</b>	Specifies the identifier, dictionary, XML data, and binary property types.

#### 3.1.4.3.5 Attributes

None.

#### 3.1.4.3.6 Groups

None.

#### 3.1.4.3.7 Attribute Groups

None.

#### 3.1.4.4 UpdateUserConfiguration

The **UpdateUserConfiguration** operation specifies the **WSDL messages** for updating user configuration objects.

The following is the **WSDL port type** definition of this operation.

```

<wsdl:operation name="UpdateUserConfiguration">
  <wsdl:input message="tns:UpdateUserConfigurationSoapIn"/>
  <wsdl:output message="tns:UpdateUserConfigurationSoapOut"/>
</wsdl:operation>

```

The following is the **WSDL** binding specification for the operation.

```

<wsdl:operation name="UpdateUserConfiguration">

```

```

    <soap:operation
      soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/UpdateUserConfiguration" />
    <wsdl:input>
      <soap:header message="tns:UpdateUserConfigurationSoapIn" part="Impersonation"
        use="literal"/>
      <soap:header message="tns:UpdateUserConfigurationSoapIn" part="MailboxCulture"
        use="literal"/>
      <soap:header message="tns:UpdateUserConfigurationSoapIn" part="RequestVersion"
        use="literal"/>
      <soap:body parts="request" use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body parts="UpdateUserConfigurationResult" use="literal" />
      <soap:header message="tns:UpdateUserConfigurationSoapOut" part="ServerVersion"
        use="literal"/>
    </wsdl:output>
  </wsdl:operation>

```

The protocol client sends a `UpdateUserConfigurationSoapIn` request WSDL message, as specified in section [3.1.4.4.1.1](#), and the protocol server responds with a `UpdateUserConfigurationSoapOut` response message, as defined in section [3.1.4.4.1.2](#).

### 3.1.4.4.1 Messages

The following table lists and describes the **WSDL message** definitions that are specific to the **UpdateUserConfiguration** operation.

Message	Description
<b>UpdateUserConfigurationSoapIn</b>	Specifies the <b>SOAP message</b> that updates a user configuration object.
<b>UpdateUserConfigurationSoapOut</b>	Specifies the SOAP message that is returned by the server in response to an <b>UpdateUserConfiguration</b> operation request.

#### 3.1.4.4.1.1 UpdateUserConfigurationSoapIn

The **UpdateUserConfigurationSoapIn WSDL message** specifies an **UpdateUserConfiguration** operation request to update a user configuration object.

```

<wsdl:message name="UpdateUserConfigurationSoapIn">
  <wsdl:part name="request" element="tns:UpdateUserConfiguration" />
  <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
  <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
  <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
</wsdl:message>

```

The **UpdateUserConfigurationSoapIn** WSDL message is the input message for the **SOAP action** `http://schemas.microsoft.com/exchange/services/2006/messages/UpdateUserConfiguration`.

The parts of the **UpdateUserConfigurationSoapIn** WSDL message are listed and described in the following table.

Part name	Element/type	Description
<b>request</b>	<b>tns:UpdateUserConfiguration</b> (section <a href="#">3.1.4.4.2.1</a> )	Specifies the <b>SOAP body</b> of a request to update a user configuration object. This part is required.



Part name	Element/type	Description
<b>Impersonation</b>	<b>t:ExchangeImpersonation</b> ([MS-OXWSCDATA] section 2.2.3.3)	Specifies a <b>SOAP header</b> that identifies the user whom the client application is impersonating. This part is optional.
<b>MailboxCulture</b>	<b>t:MailboxCulture</b> ([MS-OXWSCDATA] section 2.2.3.6)	Specifies a SOAP header that identifies the culture to be used for accessing the <b>mailbox</b> . The cultures are defined in [RFC3066]. This part is optional.
<b>RequestVersion</b>	<b>t:RequestServerVersion</b> ([MS-OXWSCDATA] section 2.2.3.9)	Specifies a SOAP header that identifies the schema version for a <b>UpdateUserConfiguration</b> operation (section 3.1.4.4) request.

### 3.1.4.4.1.2 UpdateUserConfigurationSoapOut

The **UpdateUserConfigurationSoapOut** **WSDL message** specifies the server response to an **UpdateUserConfiguration** operation request to update a user configuration object.

```
<wsdl:message name="UpdateUserConfigurationSoapOut">
  <wsdl:part name="UpdateUserConfigurationResult"
    element="tns:UpdateUserConfigurationResponse"/>
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
```

The parts of the **UpdateUserConfigurationSoapOut** WSDL message are listed and described in the following table.

Part name	Element/type	Description
<b>UpdateUserConfigurationResult</b>	<b>tns:UpdateUserConfigurationResponse</b> (section 3.1.4.4.2.2)	Specifies the <b>SOAP body</b> of the response to an <b>UpdateUserConfiguration</b> operation request.
<b>ServerVersion</b>	<b>t:ServerVersionInfo</b> ([MS-OXWSCDATA] section 2.2.3.10)	Specifies the <b>SOAP header</b> that identifies the server version for the response to an <b>UpdateUserConfiguration</b> operation request.

A successful **UpdateUserConfiguration** operation request returns an **UpdateUserConfigurationResponse** element, as specified in section 3.1.4.4.2.2, with the **ResponseClass** attribute of the **UpdateUserConfigurationResponseMessage** element set to "Success". The **ResponseCode** element of the **UpdateUserConfigurationResponseMessage** element is set to "NoError". The **UpdateUserConfigurationResponseMessage** element is a child element of the **ArrayOfResponseMessagesType** complex type, as specified in [MS-OXWSCDATA] section 2.2.4.12.

If the **UpdateUserConfiguration** operation request is not successful, it returns an **UpdateUserConfigurationResponse** element with the **ResponseClass** attribute of the **UpdateUserConfigurationResponseMessage** element set to "Error". The **ResponseCode** element of the **UpdateUserConfigurationResponseMessage** element is set to one of the common errors defined in [MS-OXWSCDATA] section 2.2.5.24.

### 3.1.4.4.2 Elements

The following table summarizes the **XML schema** element definitions that are specific to the **UpdateUserConfiguration** operation.

Element	Description
<b>UpdateUserConfiguration</b>	Specifies a request to update a user configuration object.
<b>UpdateUserConfigurationResponse</b>	Specifies the response to an <b>UpdateUserConfiguration</b> operation request.

#### 3.1.4.4.2.1 UpdateUserConfiguration Element

The **UpdateUserConfiguration** element specifies a request to update a user configuration object.

```
<xs:element name="UpdateUserConfiguration"
  type="m:UpdateUserConfigurationType"
/>
```

#### 3.1.4.4.2.2 UpdateUserConfigurationResponse Element

The **UpdateUserConfigurationResponse** element specifies the response to an **UpdateUserConfiguration** operation request (section [3.1.4.4](#)).

```
<xs:element name="UpdateUserConfigurationResponse"
  type="m:UpdateUserConfigurationResponseType"
/>
```

### 3.1.4.4.3 Complex Types

The following table summarizes the **XML schema** complex type definitions that are specific to the **UpdateUserConfiguration** operation.

Complex type	Description
<b>UpdateUserConfigurationResponseType</b>	Specifies the response to an <b>UpdateUserConfiguration</b> operation request.
<b>UpdateUserConfigurationType</b>	Specifies an <b>UpdateUserConfiguration</b> operation request.

#### 3.1.4.4.3.1 m:UpdateUserConfigurationResponseType Complex Type

The **UpdateUserConfigurationResponseType** complex type specifies the response to an **UpdateUserConfiguration** operation request (section [3.1.4.4](#)). The **UpdateUserConfigurationResponseType** complex type extends the **BaseResponseMessageType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.4.18.

```
<xs:complexType name="UpdateUserConfigurationResponseType">
  <xs:complexContent>
    <xs:extension
```

```

        base="m:BaseResponseMessageType"
      />
    </xs:complexContent>
  </xs:complexType>

```

### 3.1.4.4.3.2 m:UpdateUserConfigurationType Complex Type

The **UpdateUserConfigurationType** complex type specifies an **UpdateUserConfiguration** operation request (section [3.1.4.4](#)). The **UpdateUserConfigurationType** complex type extends the **BaseRequestType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.4.17.

```

<xs:complexType name="UpdateUserConfigurationType">
  <xs:complexContent>
    <xs:extension
      base="m:BaseRequestType"
    >
      <xs:sequence>
        <xs:element name="UserConfiguration"
          type="t:UserConfigurationType"
        />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

The following table summarizes the child element of the **UpdateUserConfigurationType** complex type.

Element	Type	Description
<b>UserConfiguration</b>	<b>t:UserConfigurationType</b> (section <a href="#">2.2.4.5</a> )	Specifies the user configuration object that contains updated data.

### 3.1.4.4.4 Simple Types

None.

### 3.1.4.4.5 Attributes

None.

### 3.1.4.4.6 Groups

None.

### 3.1.4.4.7 Attribute Groups

None.

## 3.1.5 Timer Events

None.

### **3.1.6 Other Local Events**

None.

## 4 Protocol Examples

The following examples show the request and response XML for the User Configuration Web Service Protocol operations.

### 4.1 CreateUserConfiguration Operation

The following example of a **CreateUserConfiguration** operation request, as described in section [3.1.4.1](#), shows how to form a request to create a user configuration object on the Inbox folder.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Header>
    <RequestServerVersion Version="Exchange2013"
      xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
  </soap:Header>
  <soap:Body>
    <CreateUserConfiguration
      xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
      <UserConfiguration>
        <UserConfigurationName Name="PersonalDetails"
          xmlns="http://schemas.microsoft.com/exchange/services/2006/types">
          <DistinguishedFolderId Id="inbox" />
        </UserConfigurationName>
        <Dictionary xmlns="http://schemas.microsoft.com/exchange/services/2006/types">
          <DictionaryEntry>
            <DictionaryKey>
              <Type>String</Type>
              <Value>Birthday</Value>
            </DictionaryKey>
            <DictionaryValue>
              <Type>DateTime</Type>
              <Value>2000-01-01</Value>
            </DictionaryValue>
          </DictionaryEntry>
        </Dictionary>
      </UserConfiguration>
    </CreateUserConfiguration>
  </soap:Body>
</soap:Envelope>
```

The following example shows a successful response to the **CreateUserConfiguration** operation request.

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <h:ServerVersionInfo MajorVersion="15" MinorVersion="0" MajorBuildNumber="1054"
      MinorBuildNumber="12" Version="Exchange2013"
      xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
      xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>
  </s:Header>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
```

```

<m:CreateUserConfigurationResponse
  xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
  xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
  <m:ResponseMessages>
    <m:CreateUserConfigurationResponseMessage ResponseClass="Success">
      <m:ResponseCode>NoError</m:ResponseCode>
    </m:CreateUserConfigurationResponseMessage>
  </m:ResponseMessages>
</m:CreateUserConfigurationResponse>
</s:Body>
</s:Envelope>

```

## 4.2 DeleteUserConfiguration Operation

The following example of a **DeleteUserConfiguration** operation request, as described in section [3.1.4.2](#), shows how to form a request to delete a user configuration object on the Inbox folder.

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Header>
    <RequestServerVersion Version="Exchange2013"
      xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
  </soap:Header>
  <soap:Body>
    <DeleteUserConfiguration
      xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
      <UserConfigurationName Name="PersonalDetails">
        <DistinguishedFolderId Id="inbox"
          xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
      </UserConfigurationName>
    </DeleteUserConfiguration>
  </soap:Body>
</soap:Envelope>

```

The following example shows a successful response to the **DeleteUserConfiguration** operation request.

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <h:ServerVersionInfo MajorVersion="15" MinorVersion="0" MajorBuildNumber="1054"
      MinorBuildNumber="12" Version="Exchange2013"
      xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
      xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>
  </s:Header>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <m:DeleteUserConfigurationResponse
      xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
      xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
      <m:ResponseMessages>
        <m:DeleteUserConfigurationResponseMessage ResponseClass="Success">
          <m:ResponseCode>NoError</m:ResponseCode>
        </m:DeleteUserConfigurationResponseMessage>
      </m:ResponseMessages>
    </m:DeleteUserConfigurationResponse>
  </s:Body>

```

</s:Envelope>

### 4.3 GetUserConfiguration Operation

The following example of a **GetUserConfiguration** operation request, as described in section [3.1.4.3](#), shows how to form a request to get a user configuration object on the Inbox folder.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Header>
    <RequestServerVersion Version="Exchange2013"
      xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
  </soap:Header>
  <soap:Body>
    <GetUserConfiguration
      xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
      <UserConfigurationName Name="PersonalDetails">
        <DistinguishedFolderId Id="inbox"
          xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
      </UserConfigurationName>
      <UserConfigurationProperties>All</UserConfigurationProperties>
    </GetUserConfiguration>
  </soap:Body>
</soap:Envelope>
```

The following example shows a successful response to the **GetUserConfiguration** operation request.

In this example, **Id** and **ChangeKey** attributes of the **ItemId** element, encoded with base64 encoding, have been shortened to preserve the readability.

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <h:ServerVersionInfo MajorVersion="15" MinorVersion="0" MajorBuildNumber="1054"
      MinorBuildNumber="12" Version="Exchange2013"
      xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
      xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>
  </s:Header>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <m:GetUserConfigurationResponse
      xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
      xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
      <m:ResponseMessages>
        <m:GetUserConfigurationResponseMessage ResponseClass="Success">
          <m:ResponseCode>NoError</m:ResponseCode>
          <m:UserConfiguration>
            <t:UserConfigurationName Name="PersonalDetails">
              <t:DistinguishedFolderId Id="inbox"/>
            </t:UserConfigurationName>
            <t:ItemId Id="AAMGUxYmU1Zj=" ChangeKey="CQAAAABYAUKH"/>
            <t:Dictionary>
              <t:DictionaryEntry>
                <t:DictionaryKey>
```

```

        <t:Type>String</t:Type>
        <t:Value>Birthday</t:Value>
      </t:DictionaryKey>
      <t:DictionaryValue>
        <t:Type>DateTime</t:Type>
        <t:Value>2000-01-01T00:00:00Z</t:Value>
      </t:DictionaryValue>
    </t:DictionaryEntry>
  </t:Dictionary>
</m:UserConfiguration>
</m:GetUserConfigurationResponseMessage>
</m:ResponseMessages>
</m:GetUserConfigurationResponse>
</s:Body>
</s:Envelope>

```

#### 4.4 UpdateUserConfiguration Operation

The following example of an **UpdateUserConfiguration** operation request, as described in section [3.1.4.4](#), shows how to form a request to update a user configuration object on the Inbox folder.

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Header>
    <RequestServerVersion Version="Exchange2013"
      xmlns="http://schemas.microsoft.com/exchange/services/2006/types" />
  </soap:Header>
  <soap:Body>
    <UpdateUserConfiguration
      xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
      <UserConfiguration>
        <UserConfigurationName Name="PersonalDetails"
          xmlns="http://schemas.microsoft.com/exchange/services/2006/types">
          <DistinguishedFolderId Id="inbox" />
        </UserConfigurationName>
        <Dictionary xmlns="http://schemas.microsoft.com/exchange/services/2006/types">
          <DictionaryEntry>
            <DictionaryKey>
              <Type>String</Type>
              <Value>Birthday</Value>
            </DictionaryKey>
            <DictionaryValue>
              <Type>DateTime</Type>
              <Value>1990-09-09</Value>
            </DictionaryValue>
          </DictionaryEntry>
        </Dictionary>
      </UserConfiguration>
    </UpdateUserConfiguration>
  </soap:Body>
</soap:Envelope>

```

The following example shows a successful response to the **UpdateUserConfiguration** operation request.

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">

```



```
<s:Header>
  <h:ServerVersionInfo MajorVersion="15" MinorVersion="0" MajorBuildNumber="1054"
    MinorBuildNumber="12" Version="Exchange2013"
    xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
    xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>
</s:Header>
<s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <m:UpdateUserConfigurationResponse
    xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
    xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
    <m:ResponseMessages>
      <m:UpdateUserConfigurationResponseMessage ResponseClass="Success">
        <m:ResponseCode>NoError</m:ResponseCode>
      </m:UpdateUserConfigurationResponseMessage>
    </m:ResponseMessages>
  </m:UpdateUserConfigurationResponse>
</s:Body>
</s:Envelope>
```

## **5 Security**

### **5.1 Security Considerations for Implementers**

None.

### **5.2 Index of Security Parameters**

None.

## 6 Appendix A: Full WSDL

The XML files that are listed in the following table are required in order to implement the functionality specified in this document.

File name	Description	Section
MS-OXWSUSRCFG.wsdl	Contains the WSDL for the implementation of this protocol.	6
MS-OXWSUSRCFG-messages.xsd	Contains the XML schema message definitions that are used in this protocol.	<a href="#">7.1</a>
MS-OXWSUSRCFG-types.xsd	Contains the XML schema type definitions that are used in this protocol.	<a href="#">7.2</a>

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

This section contains the contents of the MS-OXWSUSRCFG.wsdl file.

```
<?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="Exchange2016"
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"
xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
      <xs:include schemaLocation="MS-OXWSUSRCFG-messages.xsd"/>
    </xs:schema>
    <xs:schema id="types" elementFormDefault="qualified" version="Exchange2016"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns="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:import namespace="http://www.w3.org/XML/1998/namespace"/>
    </xs:schema>
  </wsdl:types>
  <wsdl:message name="CreateUserConfigurationSoapIn">
    <wsdl:part name="request" element="tns:CreateUserConfiguration"/>
    <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
    <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
    <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
  </wsdl:message>
  <wsdl:message name="CreateUserConfigurationSoapOut">
    <wsdl:part name="CreateUserConfigurationResult"
element="tns:CreateUserConfigurationResponse"/>
    <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
  </wsdl:message>
  <wsdl:message name="DeleteUserConfigurationSoapIn">
    <wsdl:part name="request" element="tns>DeleteUserConfiguration"/>
    <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
    <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
    <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
  </wsdl:message>
  <wsdl:message name="DeleteUserConfigurationSoapOut">
```

```

        <wsdl:part name="DeleteUserConfigurationResult"
element="tns:DeleteUserConfigurationResponse"/>
        <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
    </wsdl:message>
    <wsdl:message name="GetUserConfigurationSoapIn">
        <wsdl:part name="request" element="tns:GetUserConfiguration"/>
        <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
        <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
        <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
    </wsdl:message>
    <wsdl:message name="GetUserConfigurationSoapOut">
        <wsdl:part name="GetUserConfigurationResult"
element="tns:GetUserConfigurationResponse"/>
        <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
    </wsdl:message>
    <wsdl:message name="UpdateUserConfigurationSoapIn">
        <wsdl:part name="request" element="tns:UpdateUserConfiguration"/>
        <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
        <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
        <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
    </wsdl:message>
    <wsdl:message name="UpdateUserConfigurationSoapOut">
        <wsdl:part name="UpdateUserConfigurationResult"
element="tns:UpdateUserConfigurationResponse"/>
        <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
    </wsdl:message>
    <wsdl:portType name="ExchangeServicePortType">
        <wsdl:operation name="CreateUserConfiguration">
            <wsdl:input message="tns:CreateUserConfigurationSoapIn"/>
            <wsdl:output message="tns:CreateUserConfigurationSoapOut"/>
        </wsdl:operation>
        <wsdl:operation name="DeleteUserConfiguration">
            <wsdl:input message="tns:DeleteUserConfigurationSoapIn"/>
            <wsdl:output message="tns:DeleteUserConfigurationSoapOut"/>
        </wsdl:operation>
        <wsdl:operation name="GetUserConfiguration">
            <wsdl:input message="tns:GetUserConfigurationSoapIn"/>
            <wsdl:output message="tns:GetUserConfigurationSoapOut"/>
        </wsdl:operation>
        <wsdl:operation name="UpdateUserConfiguration">
            <wsdl:input message="tns:UpdateUserConfigurationSoapIn"/>
            <wsdl:output message="tns:UpdateUserConfigurationSoapOut"/>
        </wsdl:operation>
    </wsdl:portType>
    <wsdl:binding name="ExchangeServiceBinding" type="tns:ExchangeServicePortType">
        <wsdl:documentation>
            <wsi:Claim conformsTo="http://ws-i.org/profiles/basic/1.0"
xmlns:wsi="http://ws-i.org/schemas/conformanceClaim"/>
        </wsdl:documentation>
        <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
        <wsdl:operation name="CreateUserConfiguration">
            <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/CreateUserConfigurat
ion"/>
            <wsdl:input>
                <soap:header message="tns:CreateUserConfigurationSoapIn"
part="Impersonation" use="literal"/>
                <soap:header message="tns:CreateUserConfigurationSoapIn"
part="MailboxCulture" use="literal"/>
                <soap:header message="tns:CreateUserConfigurationSoapIn"
part="RequestVersion" use="literal"/>
                <soap:body parts="request" use="literal"/>
            </wsdl:input>
            <wsdl:output>
                <soap:body parts="CreateUserConfigurationResult" use="literal"/>
                <soap:header message="tns:CreateUserConfigurationSoapOut"
part="ServerVersion" use="literal"/>
            </wsdl:output>
        </wsdl:operation>

```

```

        <wsdl:operation name="DeleteUserConfiguration">
            <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/DeleteUserConfigurat
ion"/>
                <wsdl:input>
                    <soap:header message="tns:DeleteUserConfigurationSoapIn"
part="Impersonation" use="literal"/>
                    <soap:header message="tns:DeleteUserConfigurationSoapIn"
part="MailboxCulture" use="literal"/>
                    <soap:header message="tns:DeleteUserConfigurationSoapIn"
part="RequestVersion" use="literal"/>
                    <soap:body parts="request" use="literal"/>
                </wsdl:input>
                <wsdl:output>
                    <soap:body parts="DeleteUserConfigurationResult" use="literal"/>
                    <soap:header message="tns:DeleteUserConfigurationSoapOut"
part="ServerVersion" use="literal"/>
                </wsdl:output>
            </wsdl:operation>
            <wsdl:operation name="GetUserConfiguration">
                <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/GetUserConfiguration
"/>
                    <wsdl:input>
                        <soap:header message="tns:GetUserConfigurationSoapIn"
part="Impersonation" use="literal"/>
                        <soap:header message="tns:GetUserConfigurationSoapIn"
part="MailboxCulture" use="literal"/>
                        <soap:header message="tns:GetUserConfigurationSoapIn"
part="RequestVersion" use="literal"/>
                        <soap:body parts="request" use="literal"/>
                    </wsdl:input>
                    <wsdl:output>
                        <soap:body parts="GetUserConfigurationResult" use="literal"/>
                        <soap:header message="tns:GetUserConfigurationSoapOut"
part="ServerVersion" use="literal"/>
                    </wsdl:output>
                </wsdl:operation>
            <wsdl:operation name="UpdateUserConfiguration">
                <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/UpdateUserConfigurat
ion"/>
                    <wsdl:input>
                        <soap:header message="tns:UpdateUserConfigurationSoapIn"
part="Impersonation" use="literal"/>
                        <soap:header message="tns:UpdateUserConfigurationSoapIn"
part="MailboxCulture" use="literal"/>
                        <soap:header message="tns:UpdateUserConfigurationSoapIn"
part="RequestVersion" use="literal"/>
                        <soap:body parts="request" use="literal"/>
                    </wsdl:input>
                    <wsdl:output>
                        <soap:body parts="UpdateUserConfigurationResult" use="literal"/>
                        <soap:header message="tns:UpdateUserConfigurationSoapOut"
part="ServerVersion" use="literal"/>
                    </wsdl:output>
                </wsdl:operation>
            </wsdl:binding>
        </wsdl:definitions>

```

## 7 Appendix B: Full XML Schema

For ease of implementation, the following sections provide the full XML schema for this protocol.

Schema name	Prefix	Section
Messages schema	m:	<a href="#">7.1</a>
Types schema	t:	<a href="#">7.2</a>

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

### 7.1 Messages Schema

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

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

File name	Defining specification
MS-OXWSCDATA-messages.xsd	<a href="#">[MS-OXWSCDATA]</a> section 7.1

```
<?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="Exchange2016" id="messages">
  <xs:include schemaLocation="MS-OXWSCDATA-messages.xsd"/>
  <xs:import namespace="http://schemas.microsoft.com/exchange/services/2006/types"
    schemaLocation="MS-OXWSUSRCFG-types.xsd"/>
  <xs:complexType name="CreateUserConfigurationType">
    <xs:complexContent>
      <xs:extension base="m:BaseRequestType">
        <xs:sequence>
          <xs:element name="UserConfiguration"
            type="t:UserConfigurationType"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="CreateUserConfiguration" type="m:CreateUserConfigurationType"/>
  <xs:complexType name="CreateUserConfigurationResponseType">
    <xs:complexContent>
      <xs:extension base="m:BaseResponseMessageType"/>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="CreateUserConfigurationResponse"
    type="m:CreateUserConfigurationResponseType"/>
  <xs:complexType name="DeleteUserConfigurationType">
    <xs:complexContent>
      <xs:extension base="m:BaseRequestType">
        <xs:sequence>
          <xs:element name="UserConfigurationName"
            type="t:UserConfigurationNameType"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
```

```

        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
    <xs:element name="DeleteUserConfiguration" type="m:DeleteUserConfigurationType"/>
    <xs:complexType name="DeleteUserConfigurationResponseType">
      <xs:complexContent>
        <xs:extension base="m:BaseResponseMessageType"/>
      </xs:complexContent>
    </xs:complexType>
    <xs:element name="DeleteUserConfigurationResponse"
type="m:DeleteUserConfigurationResponseType"/>
    <xs:complexType name="GetUserConfigurationType">
      <xs:complexContent>
        <xs:extension base="m:BaseRequestType">
          <xs:sequence>
            <xs:element name="UserConfigurationName"
type="t:UserConfigurationNameType"/>
            <xs:element name="UserConfigurationProperties"
type="t:UserConfigurationPropertyType"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
    <xs:element name="GetUserConfiguration" type="m:GetUserConfigurationType"/>
    <xs:complexType name="GetUserConfigurationResponseMessageType">
      <xs:complexContent>
        <xs:extension base="m:ResponseMessageType">
          <xs:sequence>
            <xs:element name="UserConfiguration" type="t:UserConfigurationType"
minOccurs="0"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
    <xs:element name="GetUserConfigurationResponse" type="m:
GetUserConfigurationResponseMessageType"/>
    <xs:complexType name="UpdateUserConfigurationType">
      <xs:complexContent>
        <xs:extension base="m:BaseRequestType">
          <xs:sequence>
            <xs:element name="UserConfiguration"
type="t:UserConfigurationType"/>
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
    <xs:element name="UpdateUserConfiguration" type="m:UpdateUserConfigurationType"/>
    <xs:complexType name="UpdateUserConfigurationResponseType">
      <xs:complexContent>
        <xs:extension base="m:BaseResponseMessageType"/>
      </xs:complexContent>
    </xs:complexType>
    <xs:element name="UpdateUserConfigurationResponse"
type="m:UpdateUserConfigurationResponseType"/>
  </xs:schema>

```

## 7.2 Types Schema

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

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

File name	Defining specification
MS-OXWSFOLD-types.xsd	<a href="#">[MS-OXWSFOLD]</a> section 7.2

```

<?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="Exchange2016" id="types">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xs:include schemaLocation="MS-OXWSFOLD-types.xsd"/>
  <xs:complexType name="UserConfigurationNameType">
    <xs:complexContent>
      <xs:extension base="t:TargetFolderIdType">
        <xs:attribute name="Name" type="t:NonEmptyStringType" use="required"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:simpleType name="UserConfigurationDictionaryObjectTypesType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="DateTime"/>
      <xs:enumeration value="Boolean"/>
      <xs:enumeration value="Byte"/>
      <xs:enumeration value="String"/>
      <xs:enumeration value="Integer32"/>
      <xs:enumeration value="UnsignedInteger32"/>
      <xs:enumeration value="Integer64"/>
      <xs:enumeration value="UnsignedInteger64"/>
      <xs:enumeration value="StringArray"/>
      <xs:enumeration value="ByteArray"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="UserConfigurationDictionaryObjectType">
    <xs:sequence>
      <xs:element name="Type" type="t:UserConfigurationDictionaryObjectTypesType"/>
      <xs:element name="Value" type="xs:string" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="UserConfigurationDictionaryEntryType">
    <xs:sequence>
      <xs:element name="DictionaryKey"
type="t:UserConfigurationDictionaryObjectType"/>
      <xs:element name="DictionaryValue"
type="t:UserConfigurationDictionaryObjectType" nillable="true"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="UserConfigurationDictionaryType">
    <xs:sequence>
      <xs:element name="DictionaryEntry"
type="t:UserConfigurationDictionaryEntryType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="UserConfigurationType">
    <xs:sequence>
      <xs:element name="UserConfigurationName" type="t:UserConfigurationNameType"/>
      <xs:element name="ItemId" type="t:ItemIdType" minOccurs="0"/>
      <xs:element name="Dictionary" type="t:UserConfigurationDictionaryType"
minOccurs="0"/>
      <xs:element name="XmlData" type="xs:base64Binary" minOccurs="0"/>
      <xs:element name="BinaryData" type="xs:base64Binary" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="UserConfigurationPropertyType">
    <xs:list>
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="Id"/>
          <xs:enumeration value="Dictionary"/>

```



```
        <xs:enumeration value="XmlData"/>
        <xs:enumeration value="BinaryData"/>
        <xs:enumeration value="All"/>
    </xs:restriction>
</xs:simpleType>
</xs:list>
</xs:simpleType>
</xs:schema>
```

## 8 Appendix C: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include updates to those products.

- Microsoft Exchange Server 2010
- Microsoft Exchange Server 2013
- Microsoft Exchange Server 2016
- Microsoft Exchange Server 2019

Exceptions, if any, are noted in this section. If an update version, service pack or Knowledge Base (KB) number appears with a product name, the behavior changed in that update. The new behavior also applies to subsequent updates 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 that is 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 the product does not follow the prescription.

## 9 Change Tracking

This section identifies changes that were made to this document since the last release. Changes are classified as Major, Minor, or None.

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.
- A document revision that captures changes to protocol functionality.

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 **None** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the relevant technical content is identical to the last released version.

The changes made to this document are listed in the following table. For more information, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com).

Section	Description	Revision class
All	Updated references.	Minor

## 10 Index

### A

Abstract data model  
    [server](#) 17  
[Applicability](#) 10  
[Attribute groups](#) 16  
[Attributes](#) 16

### C

[Capability negotiation](#) 10  
[Change tracking](#) 51  
[Complex types](#) 12  
    [t:UserConfigurationDictionaryEntryType Complex Type](#) 12  
    [t:UserConfigurationDictionaryObjectType Complex Type](#) 12  
    [t:UserConfigurationDictionaryType Complex Type](#) 13  
    [t:UserConfigurationNameType Complex Type](#) 13  
    [t:UserConfigurationType Complex Type](#) 14

### D

Data model - abstract  
    [server](#) 17

### E

Events  
    [local - server](#) 36  
    [timer - server](#) 35

### F

[Fields - vendor-extensible](#) 10  
[Full WSDL](#) 43  
[Full XML schema](#) 46  
    [Messages Schema](#) 46  
    [Types Schema](#) 47

### G

[Glossary](#) 7  
[Groups](#) 16

### I

[Implementer - security considerations](#) 42  
[Index of security parameters](#) 42  
[Informative references](#) 9  
Initialization  
    [server](#) 17  
[Introduction](#) 7

### L

Local events  
    [server](#) 36

### M

Message processing  
    [server](#) 17  
Messages  
    [attribute groups](#) 16  
    [attributes](#) 16  
    [complex types](#) 12  
    [elements](#) 11  
    [enumerated](#) 11  
    [groups](#) 16  
    [namespaces](#) 11  
    [simple types](#) 15  
    [syntax](#) 11  
    [t:UserConfigurationDictionaryEntryType Complex Type complex type](#) 12  
    [t:UserConfigurationDictionaryObjectType Complex Type complex type](#) 12  
    [t:UserConfigurationDictionaryObjectTypesType Simple Type simple type](#) 15  
    [t:UserConfigurationDictionaryType Complex Type complex type](#) 13  
    [t:UserConfigurationNameType Complex Type complex type](#) 13  
    [t:UserConfigurationType Complex Type complex type](#) 14  
    [transport](#) 11

### N

[Namespaces](#) 11  
[Normative references](#) 8

### O

Operations  
    [CreateUserConfiguration](#) 17  
    [DeleteUserConfiguration](#) 22  
    [GetUserConfiguration](#) 26  
    [UpdateUserConfiguration](#) 31  
[Overview \(synopsis\)](#) 9

### P

[Parameters - security index](#) 42  
[Preconditions](#) 10  
[Prerequisites](#) 10  
[Product behavior](#) 50  
Protocol Details  
    [overview](#) 17  
  
**R**  
  
[References](#) 8  
    [informative](#) 9  
    [normative](#) 8  
[Relationship to other protocols](#) 9

### S

Security

- [implementer considerations](#) 42
- [parameter index](#) 42
- Sequencing rules
  - [server](#) 17
- Server
  - [abstract data model](#) 17
  - [CreateUserConfiguration operation](#) 17
  - [DeleteUserConfiguration operation](#) 22
  - [GetUserConfiguration operation](#) 26
  - [initialization](#) 17
  - [local events](#) 36
  - [message processing](#) 17
  - [sequencing rules](#) 17
  - [timer events](#) 35
  - [timers](#) 17
  - [UpdateUserConfiguration operation](#) 31
- [Simple types](#) 15
  - [t:UserConfigurationDictionaryObjectTypesType Simple Type](#) 15
- [Standards assignments](#) 10
- Syntax
  - [messages - overview](#) 11

## T

- [t:UserConfigurationDictionaryEntryType Complex Type complex type](#) 12
- [t:UserConfigurationDictionaryObjectType Complex Type complex type](#) 12
- [t:UserConfigurationDictionaryObjectTypesType Simple Type simple type](#) 15
- [t:UserConfigurationDictionaryType Complex Type complex type](#) 13
- [t:UserConfigurationNameType Complex Type complex type](#) 13
- [t:UserConfigurationType Complex Type complex type](#) 14
- Timer events
  - [server](#) 35
- Timers
  - [server](#) 17
- [Tracking changes](#) 51
- [Transport](#) 11
- Types
  - [complex](#) 12
  - [simple](#) 15

## V

- [Vendor-extensible fields](#) 10
- [Versioning](#) 10

## W

- [WSDL](#) 43

## X

- [XML schema](#) 46
  - [Messages Schema](#) 46
  - [Types Schema](#) 47