

# [MS-OXWSUSRCFG]: User Configuration Web Service Protocol Specification

---

## Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft's Open Specification Promise (available here: <http://www.microsoft.com/interop/osp>) or the Community Promise (available here: <http://www.microsoft.com/interop/cp/default.msp>). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting [iplg@microsoft.com](mailto:iplg@microsoft.com).
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.
- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

**Reservation of Rights.** All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

**Tools.** The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

## Revision Summary

Date	Revision History	Revision Class	Comments
07/15/2009	1.0	Major	Initial Availability.
11/04/2009	1.0.1	Editorial	Revised and edited the technical content.
02/10/2010	1.2.0	Minor	Updated the technical content.
05/05/2010	1.2.1	Editorial	Revised and edited the technical content.
08/04/2010	1.3	Minor	Clarified the meaning of the technical content.

# Contents

<b>1 Introduction</b>	<b>6</b>
1.1 Glossary	6
1.2 References	6
1.2.1 Normative References	6
1.2.2 Informative References	7
1.3 Overview	7
1.4 Relationship to Other Protocols	8
1.5 Prerequisites/Preconditions	8
1.6 Applicability Statement	8
1.7 Versioning and Capability Negotiation	8
1.8 Vendor-Extensible Fields	8
1.9 Standards Assignments	9
<b>2 Messages</b>	<b>10</b>
2.1 Transport	10
2.2 Common Message Syntax	10
2.2.1 Namespaces	10
2.2.2 Simple Types	10
2.2.2.1 t:UserConfigurationDictionaryObjectTypesType Simple Type	10
2.2.2.2 t:UserConfigurationPropertyType Simple Type	12
2.2.3 Complex Types	12
2.2.3.1 t:UserConfigurationDictionaryEntryType Complex Type	12
2.2.3.2 t:UserConfigurationDictionaryObjectType Complex Type	13
2.2.3.3 t:UserConfigurationDictionaryType Complex Type	13
2.2.3.4 t:UserConfigurationNameType Complex Type	14
2.2.3.5 t:UserConfigurationType Complex Type	14
2.2.4 Elements	15
2.2.5 Attributes	15
2.2.6 Groups	15
2.2.7 Attribute Groups	15
2.2.8 Messages	15
<b>3 Protocol Details</b>	<b>16</b>
3.1 ExchangeServicePortType Server Details	16
3.1.1 Abstract Data Model	16
3.1.2 Timers	16
3.1.3 Initialization	16
3.1.4 Message Processing Events and Sequencing	16
3.1.4.1 CreateUserConfiguration	16
3.1.4.1.1 Simple Types	17
3.1.4.1.2 Complex Types	17
3.1.4.1.2.1 m:CreateUserConfigurationResponseType Complex Type	17
3.1.4.1.2.2 m:CreateUserConfigurationType Complex Type	17
3.1.4.1.3 Elements	18
3.1.4.1.3.1 CreateUserConfiguration Element	18
3.1.4.1.3.2 CreateUserConfigurationResponse Element	18
3.1.4.1.4 Attributes	18
3.1.4.1.5 Groups	18
3.1.4.1.6 Attribute Groups	18
3.1.4.1.7 Messages	19

3.1.4.1.7.1	CreateUserConfigurationSoapIn.....	19
3.1.4.1.7.2	CreateUserConfigurationSoapOut.....	19
3.1.4.2	DeleteUserConfiguration.....	19
3.1.4.2.1	Simple Types.....	20
3.1.4.2.2	Complex Types.....	20
3.1.4.2.2.1	m>DeleteUserConfigurationResponseType Complex Type.....	20
3.1.4.2.2.2	m>DeleteUserConfigurationType Complex Type.....	20
3.1.4.2.3	Elements.....	21
3.1.4.2.3.1	DeleteUserConfiguration Element.....	21
3.1.4.2.3.2	DeleteUserConfigurationResponse Element.....	21
3.1.4.2.4	Attributes.....	21
3.1.4.2.5	Groups.....	21
3.1.4.2.6	Attribute Groups.....	21
3.1.4.2.7	Messages.....	21
3.1.4.2.7.1	DeleteUserConfigurationSoapIn.....	21
3.1.4.2.7.2	DeleteUserConfigurationSoapOut.....	22
3.1.4.3	GetUserConfiguration.....	22
3.1.4.3.1	Simple Types.....	22
3.1.4.3.2	Complex Types.....	22
3.1.4.3.2.1	t:GetUserConfigurationResponseMessageType Complex Type.....	23
3.1.4.3.2.2	m:GetUserConfigurationResponseType Complex Type.....	23
3.1.4.3.2.3	m:GetUserConfigurationType Complex Type.....	23
3.1.4.3.3	Elements.....	24
3.1.4.3.3.1	GetUserConfiguration Element.....	24
3.1.4.3.3.2	GetUserConfigurationResponse Element.....	24
3.1.4.3.4	Attributes.....	24
3.1.4.3.5	Groups.....	24
3.1.4.3.6	Attribute Groups.....	25
3.1.4.3.7	Messages.....	25
3.1.4.3.7.1	GetUserConfigurationSoapIn.....	25
3.1.4.3.7.2	GetUserConfigurationSoapOut.....	25
3.1.4.4	UpdateUserConfiguration.....	25
3.1.4.4.1	Simple Types.....	26
3.1.4.4.2	Complex Types.....	26
3.1.4.4.2.1	m:UpdateUserConfigurationResponseType Complex Type.....	26
3.1.4.4.2.2	m:UpdateUserConfigurationType Complex Type.....	26
3.1.4.4.3	Elements.....	27
3.1.4.4.3.1	UpdateUserConfiguration Element.....	27
3.1.4.4.3.2	UpdateUserConfigurationResponse Element.....	27
3.1.4.4.4	Attributes.....	27
3.1.4.4.5	Groups.....	27
3.1.4.4.6	Attribute Groups.....	27
3.1.4.4.7	Messages.....	27
3.1.4.4.7.1	UpdateUserConfigurationSoapIn.....	27
3.1.4.4.7.2	UpdateUserConfigurationSoapOut.....	28
3.1.5	Timer Events.....	28
3.1.6	Other Local Events.....	28
3.2	Client Details.....	28
3.2.1	Abstract Data Model.....	28
3.2.2	Timers.....	28
3.2.3	Initialization.....	28
3.2.4	Message Processing Events and Sequencing.....	28
3.2.5	Timer Events.....	28

3.2.6 Other Local Events .....	29
<b>4 Protocol Examples .....</b>	<b>30</b>
<b>5 Security .....</b>	<b>31</b>
5.1 Security Considerations for Implementers .....	31
5.2 Index of Security Parameters .....	31
<b>6 Appendix A: Full WSDL .....</b>	<b>32</b>
6.1 WSDL.....	32
6.2 Messages Schema.....	35
6.3 Types Schema.....	37
<b>7 Appendix B: Product Behavior .....</b>	<b>39</b>
<b>8 Change Tracking.....</b>	<b>40</b>
<b>9 Index .....</b>	<b>42</b>

# 1 Introduction

This document specifies the User Configuration Web Service protocol. The User Configuration Web Service protocol is used to access user configuration objects. Clients use the SOAP protocol [\[SOAP1.1\]](#) to contact the user configuration service.

## 1.1 Glossary

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

**folder associated item (FAI)**  
**Hypertext Transfer Protocol (HTTP)**  
**Hypertext Transfer Protocol over Secure Socket Layers (HTTPS)**  
**property**  
**SOAP body**  
**SOAP message**  
**Web Services Description Language (WSDL)**  
**WSDL message**  
**WSDL port type**  
**XML**  
**XML namespace**  
**XML schema**

The following terms are specific to this document:

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

## 1.2 References

### 1.2.1 Normative References

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

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

[MS-OXWSFOLD] Microsoft Corporation, "[Folders and Folder Permissions Web Service Protocol Specification](#)", July 2009.

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

[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, <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

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

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

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

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

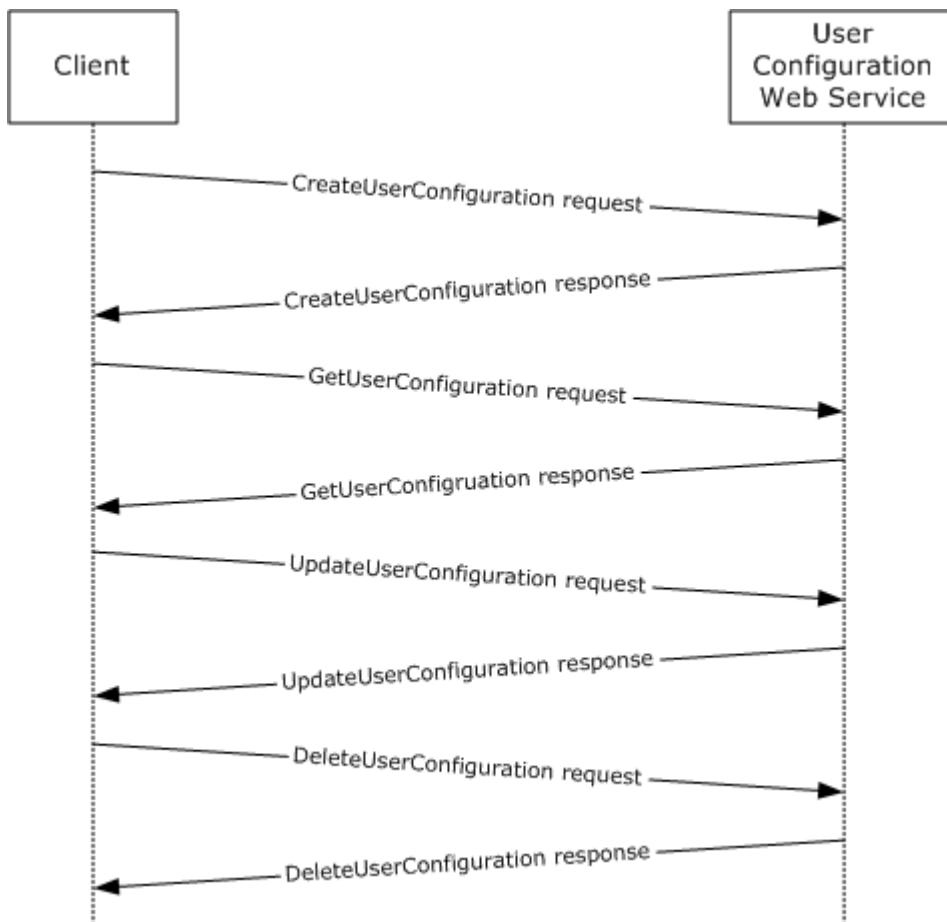
## 1.2.2 Informative References

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

## 1.3 Overview

The 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 items (FAIs)**.

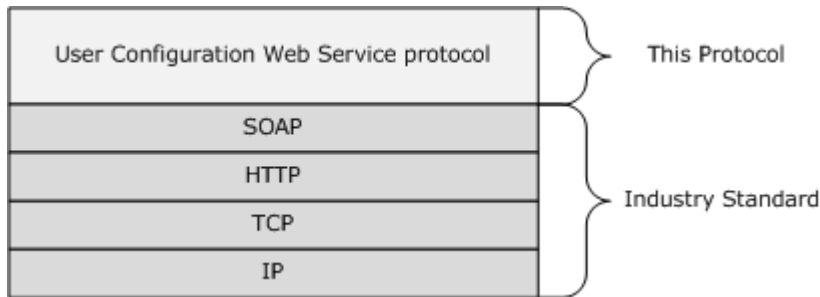
The following figure shows the User Configuration Web Service request/response messages.



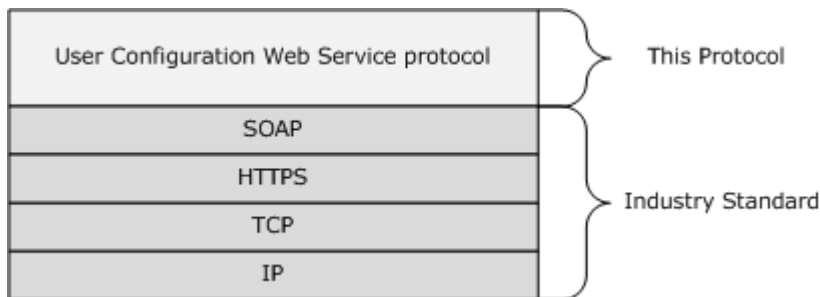
**Figure 1: User Configuration Web service request/response messages**

## 1.4 Relationship to Other Protocols

The User Configuration Web service protocol uses SOAP over **HTTP** and SOAP over **HTTPS**, as shown in the following layering diagrams.



**Figure 2: User Configuration Web Service protocol HTTP stack**



**Figure 3: User Configuration Web Service protocol HTTPS stack**

## 1.5 Prerequisites/Preconditions

None.

## 1.6 Applicability Statement

The protocol specified in this document is applicable to environments that create, delete, get, or update folder associated items (FAIs).

## 1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

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

## 1.8 Vendor-Extensible Fields

None.



## 1.9 Standards Assignments

None.

## 2 Messages

### 2.1 Transport

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

### 2.2 Common Message Syntax

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

#### 2.2.1 Namespaces

This specification defines and references various **XML namespaces** by using the mechanisms specified in [\[XMLNS\]](#). Although this specification associates a specific XML namespace prefix 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>	[MS-OXWSUSRCFG]
t	<a href="http://schemas.microsoft.com/exchange/services/2006/types">http://schemas.microsoft.com/exchange/services/2006/types</a>	[MS-OXWSUSRCFG]
xs	<a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a>	<a href="#">[XMLSCHEMA1]</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>	[MS-OXWSUSRCFG]
targetNamespace	<a href="http://schemas.microsoft.com/exchange/services/2006/messages">http://schemas.microsoft.com/exchange/services/2006/messages</a>	[MS-OXWSUSRCFG]

#### 2.2.2 Simple Types

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

```

/>
<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>

```

## Enumeration

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

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

### 2.2.2.2 t:UserConfigurationPropertyType Simple Type

The [UserConfigurationPropertyType](#) simple type specifies the property types to get in a [GetUserConfiguration](#) operation (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>
```

#### Enumeration

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

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

### 2.2.3 Complex Types

#### 2.2.3.1 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>

```

#### Child Elements

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

### 2.2.3.2 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>

```

#### Child Elements

Element	Type	Description
Type	<a href="#">t:UserConfigurationDictionaryObjectTypesType</a>	Specifies the dictionary object type.
Value	xs:string	Specifies the dictionary object value as a string.

### 2.2.3.3 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>

```

#### Child Elements

Element	Type
DictionaryEntry	<a href="#">t:UserConfigurationDictionaryEntryType</a>

#### 2.2.3.4 t:UserConfigurationNameType Complex Type

The [UserConfigurationNameType](#) complex type specifies the name and the associated folder for a user configuration object.

```

<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>

```

#### Attributes

Name	Type	Description
Name	<a href="#">t:NonEmptyStringType</a>	Specifies the name of a user configuration object.

#### 2.2.3.5 t:UserConfigurationType Complex Type

The [UserConfigurationType](#) complex types 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>

```

## Child Elements

Element	Type	Description
UserConfigurationName	<a href="#">t:UserConfigurationNameType</a>	Specifies the user configuration object name. The user configuration object name MUST be present.
ItemId	<a href="#">t:ItemIdType</a>	Specifies the user configuration object item identifier.
Dictionary	<a href="#">t:UserConfigurationDictionaryType</a>	Specifies a set of dictionary property entries.
XmlData	xs:base64Binary	Specifies XML data property content.
BinaryData	xs:base64Binary	Specifies binary data property content.

### 2.2.4 Elements

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

### 2.2.5 Attributes

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

### 2.2.6 Groups

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

### 2.2.7 Attribute Groups

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

### 2.2.8 Messages

This specification does not define any common WSDL message 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 port type.

Operation	Description
<a href="#">CreateUserConfiguration</a>	Specifies the WSDL messages for creating user configuration objects.
<a href="#">DeleteUserConfiguration</a>	Specifies the WSDL messages for deleting user configuration objects.
<a href="#">GetUserConfiguration</a>	Specifies the WSDL messages for getting user configuration objects.
<a href="#">UpdateUserConfiguration</a>	Specifies the WSDL messages for updating user configuration objects.

#### 3.1.1 Abstract Data Model

The User Configuration Web Service protocol is a stateless protocol.

#### 3.1.2 Timers

None.

#### 3.1.3 Initialization

None.

#### 3.1.4 Message Processing Events and Sequencing

This protocol includes the operations that are listed in the following table. The operations do not have any sequencing rules.

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



```

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

```

#### Request

Message Format	Description
<a href="#">tns:CreateUserConfigurationSoapIn</a>	Specifies the <b>SOAP message</b> that is sent to the server to create a user configuration object.

#### Response

Message Format	Description
<a href="#">tns:CreateUserConfigurationSoapOut</a>	Specifies the SOAP message that is sent from the server after an attempt to create a user configuration object.

### 3.1.4.1.1 Simple Types

This specification does not define any specific XML schema simple type definitions for this operation.

### 3.1.4.1.2 Complex Types

The following XML schema complex type definitions are specific to this operation.

#### 3.1.4.1.2.1 m:CreateUserConfigurationResponseType Complex Type

The [CreateUserConfigurationResponseType](#) complex type specifies a response for the [CreateUserConfiguration](#) operation (section [3.1.4.1](#)).

```

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

```

#### 3.1.4.1.2.2 m:CreateUserConfigurationType Complex Type

The [CreateUserConfigurationType](#) complex type specifies a [CreateUserConfiguration](#) operation (section [3.1.4.1](#)) request.

```

<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>

```

#### Child Elements

Element	Type	Description
UserConfiguration	<a href="#">t:UserConfigurationType</a>	Specifies the user configuration object to create.

### 3.1.4.1.3 Elements

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

#### 3.1.4.1.3.1 CreateUserConfiguration Element

The [tns:CreateUserConfiguration](#) element specifies a request to create a user configuration object.

```

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

```

#### 3.1.4.1.3.2 CreateUserConfigurationResponse Element

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

```

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

```

#### 3.1.4.1.4 Attributes

This specification does not define any specific XML schema attribute definitions for this operation.

#### 3.1.4.1.5 Groups

This specification does not define any specific XML schema group definitions for this operation.

#### 3.1.4.1.6 Attribute Groups

This specification does not define any specific XML schema attribute group definitions for this operation.

### 3.1.4.1.7 Messages

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

#### 3.1.4.1.7.1 CreateUserConfigurationSoapIn

The [CreateUserConfigurationSoapIn](#) message contains four parts.

Part Name	Element/Type	Description
request	<a href="#">tns:CreateUserConfiguration</a>	Specifies the request.
Impersonation	<a href="#">t:ExchangeImpersonation</a>	Specifies an account to impersonate.
MailboxCulture	<a href="#">t:MailboxCulture</a>	Specifies the culture to use for accessing the server. The cultures are defined by <a href="#">[RFC3066]</a> .
RequestVersion	<a href="#">t:RequestServerVersion</a>	Specifies the schema version for the <a href="#">CreateUserConfiguration</a> operation (section <a href="#">3.1.4.1</a> ) request.

#### 3.1.4.1.7.2 CreateUserConfigurationSoapOut

The [CreateUserConfigurationSoapOut](#) message contains two parts.

Part Name	Element/Type	Description
CreateUserConfigurationResult	<a href="#">tns:CreateUserConfigurationResponse</a>	Specifies the response.
ServerVersion	<a href="#">t:ServerVersionInfo</a>	Specifies the server version for the response.

### 3.1.4.2 DeleteUserConfiguration

The [DeleteUserConfiguration](#) operation specifies the WSDL messages for deleting user configuration objects.

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

Request

Message Format	Description
<a href="#">tns:DeleteUserConfigurationSoapIn</a>	Specifies the SOAP message that is sent to the server to delete a user configuration object.

Response

Message Format	Description
<a href="#">tns:DeleteUserConfigurationSoapOut</a>	Specifies the SOAP message that is sent from the server after an attempt to delete a user configuration object.

### 3.1.4.2.1 Simple Types

This specification does not define any specific XML schema simple type definitions for this operation.

### 3.1.4.2.2 Complex Types

The following XML schema complex type definitions are specific to this operation.

#### 3.1.4.2.2.1 m:DeleteUserConfigurationResponseType Complex Type

The [m:DeleteUserConfigurationResponseType](#) complex type specifies a response for the [DeleteUserConfiguration](#) operation (section [3.1.4.2](#)).

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

#### 3.1.4.2.2.2 m:DeleteUserConfigurationType Complex Type

The [DeleteUserConfigurationType](#) complex type specifies a [DeleteUserConfiguration](#) operation (section [3.1.4.2](#)) request.

```
<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>
```

#### Child Elements

Element	Type	Description
UserConfigurationName	<a href="#">t:UserConfigurationNameType</a>	Specifies the name of the user configuration object.

### 3.1.4.2.3 Elements

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

#### 3.1.4.2.3.1 DeleteUserConfiguration Element

The [DeleteUserConfiguration](#) element specifies a request to delete a user configuration object.

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

#### 3.1.4.2.3.2 DeleteUserConfigurationResponse Element

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

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

### 3.1.4.2.4 Attributes

This specification does not define any specific XML schema attribute definitions for this operation.

### 3.1.4.2.5 Groups

This specification does not define any specific XML schema group definitions for this operation.

### 3.1.4.2.6 Attribute Groups

This specification does not define any specific XML schema attribute group definitions for this operation.

### 3.1.4.2.7 Messages

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

#### 3.1.4.2.7.1 DeleteUserConfigurationSoapIn

The [DeleteUserConfigurationSoapIn](#) message contains the four parts listed in the following table.

Part Name	Element/Type	Description
request	<a href="#">tns:DeleteUserConfiguration</a>	Specifies the request.
Impersonation	<a href="#">t:ExchangeImpersonation</a>	Specifies an account to impersonate.
MailboxCulture	<a href="#">t:MailboxCulture</a>	Specifies the culture to use for accessing the server. The cultures are defined by <a href="http://www.ietf.org/rfc/rfc3066.txt">http://www.ietf.org/rfc/rfc3066.txt</a> .

Part Name	Element/Type	Description
RequestVersion	<a href="#">t:RequestServerVersion</a>	Specifies the schema version for the <a href="#">DeleteUserConfiguration</a> operation (section <a href="#">3.1.4.2</a> ) request.

### 3.1.4.2.7.2 DeleteUserConfigurationSoapOut

The [DeleteUserConfigurationSoapOut](#) message contains the two parts described in the following table.

Part Name	Element/Type	Description
DeleteUserConfigurationResult	<a href="#">tns:DeleteUserConfigurationResponse</a>	Specifies the response.
ServerVersion	<a href="#">t:ServerVersionInfo</a>	Specifies the server version for the response.

### 3.1.4.3 GetUserConfiguration

The [GetUserConfiguration](#) operation specifies the WSDL messages that are used to get user configuration objects.

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

Request

Message Format	Description
<a href="#">tns:GetUserConfigurationSoapIn</a>	Specifies the SOAP message that is sent to the server to get a user configuration object.

Response

Message Format	Description
<a href="#">tns:GetUserConfigurationSoapOut</a>	Specifies the SOAP message that is sent from the server after an attempt to get a user configuration object.

#### 3.1.4.3.1 Simple Types

This specification does not define any specific XML schema simple type definitions for this operation.

#### 3.1.4.3.2 Complex Types

The following XML schema complex type definitions are specific to this operation.

### 3.1.4.3.2.1 t:GetUserConfigurationResponseMessageType Complex Type

The [GetUserConfigurationResponseMessageType](#) complex type specifies user configuration objects that are returned in a response.

```
<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>
```

#### Child Elements

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

### 3.1.4.3.2.2 m:GetUserConfigurationResponseType Complex Type

The [GetUserConfigurationResponseType](#) complex type specifies the response for the [GetUserConfiguration](#) operation (section [3.1.4.3](#)).

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

### 3.1.4.3.2.3 m:GetUserConfigurationType Complex Type

The [GetUserConfigurationType](#) complex type specifies a [GetUserConfiguration](#) operation (section [3.1.4.3](#)) request.

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

```

## Child Elements

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

### 3.1.4.3.3 Elements

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

#### 3.1.4.3.3.1 GetUserConfiguration Element

The [GetUserConfiguration](#) element specifies a request to get a user configuration object.

```

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

```

#### 3.1.4.3.3.2 GetUserConfigurationResponse Element

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

```

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

```

#### 3.1.4.3.4 Attributes

This specification does not define any specific XML schema attribute definitions for this operation.

#### 3.1.4.3.5 Groups

This specification does not define any specific XML schema group definitions for this operation.



### 3.1.4.3.6 Attribute Groups

This specification does not define any specific XML schema attribute group definitions for this operation.

### 3.1.4.3.7 Messages

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

#### 3.1.4.3.7.1 GetUserConfigurationSoapIn

The [GetUserConfigurationSoapIn](#) message contains four parts.

Part Name	Element/Type	Description
request	<a href="#">tns:GetUserConfiguration</a>	Specifies the request.
Impersonation	<a href="#">t:ExchangeImpersonation</a>	Specifies an account to impersonate.
MailboxCulture	<a href="#">t:MailboxCulture</a>	Specifies the culture to use for accessing the server. The cultures are defined by <a href="#">[RFC3066]</a> .
RequestVersion	<a href="#">t:RequestServerVersion</a>	Specifies the schema version for the <a href="#">GetUserConfiguration</a> operation (section <a href="#">3.1.4.3</a> ) request.

#### 3.1.4.3.7.2 GetUserConfigurationSoapOut

The [GetUserConfigurationSoapOut](#) message contains two parts.

Part Name	Element/Type	Description
GetUserConfigurationResult	<a href="#">tns:GetUserConfigurationResponse</a>	Specifies the response.
ServerVersion	<a href="#">t:ServerVersionInfo</a>	Specifies the server version for the response.

### 3.1.4.4 UpdateUserConfiguration

The [UpdateUserConfiguration](#) operation specifies the WSDL messages for updating user configuration objects.

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

Request

Message Format	Description
<a href="#">tns:UpdateUserConfigurationSoapIn</a>	Specifies the SOAP message that is sent to the server to update a user configuration object.

Response

Message Format	Description
<a href="#">tns:UpdateUserConfigurationSoapOut</a>	Specifies the SOAP message that is sent from the server after an attempt to update a user configuration object.

#### 3.1.4.4.1 Simple Types

This specification does not define any specific XML schema simple type definitions for this operation.

#### 3.1.4.4.2 Complex Types

The following XML schema complex type definitions are specific to this operation.

##### 3.1.4.4.2.1 m:UpdateUserConfigurationResponseType Complex Type

The [UpdateUserConfigurationResponseType](#) complex type specifies the response object for the [UpdateUserConfiguration](#) operation (section [3.1.4.4](#)).

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

##### 3.1.4.4.2.2 m:UpdateUserConfigurationType Complex Type

The [UpdateUserConfigurationType](#) complex type specifies an [UpdateUserConfiguration](#) operation (section [3.1.4.4](#)) request.

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

Child Elements

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

### 3.1.4.4.3 Elements

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

#### 3.1.4.4.3.1 UpdateUserConfiguration Element

The [UpdateUserConfiguration](#) element specifies a request to update a user configuration object.

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

#### 3.1.4.4.3.2 UpdateUserConfigurationResponse Element

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

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

#### 3.1.4.4.4 Attributes

This specification does not define any specific XML schema attribute definitions for this operation.

#### 3.1.4.4.5 Groups

This specification does not define any specific XML schema group definitions for this operation.

#### 3.1.4.4.6 Attribute Groups

This specification does not define any specific XML schema attribute group definitions for this operation.

#### 3.1.4.4.7 Messages

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

##### 3.1.4.4.7.1 UpdateUserConfigurationSoapIn

The [UpdateUserConfigurationSoapIn](#) message contains four parts.

Part Name	Element/Type	Description
request	<a href="#">tns:UpdateUserConfiguration</a>	Specifies the request.

Part Name	Element/Type	Description
Impersonation	<a href="#">t:ExchangeImpersonation</a>	Specifies an account to impersonate.
MailboxCulture	<a href="#">t:MailboxCulture</a>	Specifies the culture to use for accessing the server. The cultures are defined by <a href="#">[RFC3066]</a> .
RequestVersion	<a href="#">t:RequestServerVersion</a>	Specifies the schema version for the <a href="#">UpdateUserConfiguration</a> operation (section <a href="#">3.1.4.4</a> ) request.

### 3.1.4.4.7.2 UpdateUserConfigurationSoapOut

The [UpdateUserConfigurationSoapOut](#) message contains two parts.

Part Name	Element/Type	Description
UpdateUserConfigurationResult	<a href="#">tns:UpdateUserConfigurationResponse</a>	Specifies the response.
ServerVersion	<a href="#">t:ServerVersionInfo</a>	Specifies the server version for the response.

### 3.1.5 Timer Events

None.

### 3.1.6 Other Local Events

None.

## 3.2 Client Details

None.

### 3.2.1 Abstract Data Model

None.

### 3.2.2 Timers

None.

### 3.2.3 Initialization

None.

### 3.2.4 Message Processing Events and Sequencing

None.

### 3.2.5 Timer Events

None.

### 3.2.6 Other Local Events

None.

## 4 Protocol Examples

None.

## **5 Security**

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

The User Configuration Web Service protocol does not use any additional security mechanisms.

### **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. The contents of each file are included in this section.

File name	Description	Section
MS-OXWSUSRCFG.wsdl	Contains the WSDL for the implementation of this protocol.	<a href="#">6.1</a>
MS-OXWSUSRCFG-messages.xsd	Contains the XML schema message definitions that are used in this protocol.	<a href="#">6.2</a>
MS-OXWSUSRCFG-types.xsd	Contains the XML schema type definitions that are used in this protocol.	<a href="#">6.3</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.

### 6.1 WSDL

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="Exchange2010"
xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
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="Exchange2010"
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: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/GetUserConfiguratio
n"/>
        <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: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:definitions>

```

## 6.2 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 6.2.

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
  xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
  xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
  elementFormDefault="qualified" version="Exchange2010" id="messages">
  <xs: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: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:complexType name="GetUserConfigurationResponseType">
        <xs:complexContent>
            <xs:extension base="m:BaseResponseMessageType"/>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="GetUserConfigurationResponse"
type="m:GetUserConfigurationResponseType"/>
    <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>

```

### 6.3 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 Protocol
MS-OXWSFOLD-types.xsd	<a href="#">[MS-OXWSFOLD]</a> section 6.

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types"
elementFormDefault="qualified" version="Exchange2010" id="types">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xs:include schemaLocation="MS-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: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>

```

## 7 Appendix B: Product Behavior

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

- Microsoft® Exchange Server 2010

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

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

## 8 Change Tracking

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

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

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

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

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

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

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

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

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



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

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

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

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

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

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
<a href="#">1.2.1 Normative References</a>	55751 Moved [MS-OXGLOS] from Normative References section to Informative References section.	N	Content update.
<a href="#">1.2.1 Normative References</a>	57730 Removed references to [RFC2396], [RFC2616], and [RFC2818].	N	Content removed.

## 9 Index

### A

Abstract data model  
[server](#) 16  
[Applicability](#) 8  
[Attribute groups](#) 15

### C

[Capability negotiation](#) 8  
[Change tracking](#) 40  
Client

[overview](#) 16

Complex types

[t:UserConfigurationDictionaryEntryType](#) 12  
[t:UserConfigurationDictionaryObjectType](#) 13  
[t:UserConfigurationDictionaryType](#) 13  
[t:UserConfigurationNameType](#) 14  
[t:UserConfigurationType](#) 14

### D

Data model - abstract  
[server](#) 16

### E

[ExchangeServicePortType server](#) 16

### F

[Full WSDL](#) 32

### G

[Glossary](#) 6  
[Groups](#) 15

### I

[Implementer - security considerations](#) 31  
[Introduction](#) 6

### M

Message processing  
[server](#) 16

Messages

[attribute groups](#) 15  
[attributes](#) 15  
[elements](#) 15  
[groups](#) 15  
[namespaces](#) 10  
[syntax](#) 10  
[t:UserConfigurationDictionaryEntryType complex type](#) 12  
[t:UserConfigurationDictionaryObjectType complex type](#) 13

[t:UserConfigurationDictionaryObjectTypesType simple type](#) 10  
[t:UserConfigurationDictionaryType complex type](#) 13  
[t:UserConfigurationNameType complex type](#) 14  
[t:UserConfigurationPropertyType simple type](#) 12  
[t:UserConfigurationType complex type](#) 14  
[transport](#) 10

### N

[Namespaces](#) 10

### O

[Overview \(synopsis\)](#) 7

### P

[Product behavior](#) 39

### R

References

[informative](#) 7  
[normative](#) 6  
[overview](#) 6

[Relationship to other protocols](#) 8

### S

Security

[implementer considerations](#) 31

Sequencing rules

[server](#) 16

Server

[abstract data model](#) 16  
[ExchangeServicePortType](#) 16  
[message processing](#) 16  
[overview](#) 16  
[sequencing rules](#) 16

[Simple types](#)

[t:UserConfigurationDictionaryObjectTypesType](#) 10  
[Simple types t:UserConfigurationPropertyType](#) 12

Syntax

[messages - overview](#) 10

### T

[t:UserConfigurationDictionaryEntryType complex type](#) 12  
[t:UserConfigurationDictionaryObjectType complex type](#) 13  
[t:UserConfigurationDictionaryObjectTypesType simple type](#) 10  
[t:UserConfigurationDictionaryType complex type](#) 13  
[t:UserConfigurationNameType complex type](#) 14  
[t:UserConfigurationPropertyType simple type](#) 12

[t:UserConfigurationType complex type](#) 14  
[Tracking changes](#) 40  
[Transport](#) 10

## **V**

[Versioning](#) 8

## **W**

[WSDL](#) 32