

[MS-OXWSGTRM]: Get Rooms List 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.
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.
- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

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

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

Revision Summary

Date	Revision History	Revision Class	Comments
07/15/2009	1.0	Major	Initial Availability.
11/04/2009	1.1.0	Minor	Updated the technical content.
02/10/2010	1.1.0	None	Version 1.1.0 release
05/05/2010	1.1.1	Editorial	Revised and edited the technical content.

Table of Contents

1 Introduction	5
1.1 Glossary	5
1.2 References	5
1.2.1 Normative References	5
1.2.2 Informative References	6
1.3 Overview	6
1.4 Relationship to Other Protocols	6
1.5 Prerequisites/Preconditions	7
1.6 Applicability Statement	7
1.7 Versioning and Capability Negotiation	7
1.8 Vendor-Extensible Fields	7
1.9 Standards Assignments	7
2 Messages	8
2.1 Transport	8
2.2 Common Message Syntax	8
2.2.1 Namespaces	8
2.2.2 Simple Types	8
2.2.3 Complex Types	8
2.2.4 Elements	8
2.2.5 Attributes	8
2.2.6 Groups	8
2.2.7 Attribute Groups	9
2.2.8 Message Syntax	9
3 Protocol Details	10
3.1 ExchangeServicePortType Server Details	10
3.1.1 Abstract Data Model	10
3.1.2 Timers	10
3.1.3 Initialization	10
3.1.4 Message Processing Events and Sequencing	10
3.1.4.1 GetRoomLists	10
3.1.4.1.1 GetRoomLists Complex Types	11
3.1.4.1.1.1 m:GetRoomListsResponseMessageType Complex Type	11
3.1.4.1.1.2 m:GetRoomListsType Complex Type	11
3.1.4.1.2 GetRoomLists Elements	11
3.1.4.1.2.1 GetRoomLists Element	12
3.1.4.1.2.2 GetRoomListsResponse Element	12
3.1.4.1.3 GetRoomLists Messages	12
3.1.4.1.3.1 tns:GetRoomListsSoapIn	12
3.1.4.1.3.2 tns:GetRoomListsSoapOut	13
3.1.4.2 Get Rooms	13
3.1.4.2.1 GetRooms Complex Types	13
3.1.4.2.1.1 m:GetRoomsResponseMessageType Complex Type	13
3.1.4.2.1.2 m:GetRoomsType Complex Type	14
3.1.4.2.1.3 t:ArrayOfRoomsType Complex Type	14
3.1.4.2.1.4 t:DirectoryEntryType Complex Type	15
3.1.4.2.1.5 t:RoomType Complex Type	15
3.1.4.2.2 GetRooms Elements	15
3.1.4.2.2.1 tns:GetRooms Element	16

3.1.4.2.2.2	tns:GetRoomsResponse Element	16
3.1.4.2.3	GetRooms Messages	16
3.1.4.2.3.1	tns:GetRoomsSoapIn	16
3.1.4.2.3.2	tns:GetRoomsSoapOut	16
3.1.5	Timer Events	17
3.1.6	Other Local Events	17
3.2	Client Details	17
3.2.1	Client Abstract Data Model	17
3.2.2	Client Timers	17
3.2.3	Client Initialization	17
3.2.4	Client Message Processing Events and Sequencing	17
3.2.5	Client Timer Events	17
3.2.6	Client Other Local Events	17
4	Protocol Examples	18
5	Security	19
5.1	Security Considerations for Implementors	19
5.2	Index of Security Parameters	19
6	Appendix A: Full WSDL	20
6.1	WSDL	20
6.2	Types Schema	22
6.3	Messages Schema	22
7	Appendix B: Product Behavior	24
8	Change Tracking	25
9	Index	27

1 Introduction

This document specifies the Get Rooms List Web Service protocol, which sends the request-response messages for retrieving meeting room information.

1.1 Glossary

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

SOAP body
SOAP fault
SOAP header
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. 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-OXGLOS] Microsoft Corporation, "[Exchange Server Protocols Master Glossary](#)", April 2008.

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

[MS-OXWSCORE] Microsoft Corporation, "[Core Items Web Service Protocol Specification](#)", July 2009.

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

[RFC2396] Berners-Lee, T., Fielding, R., and Masinter, L., "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998, <http://www.ietf.org/rfc/rfc2396.txt>

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

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

[RFC3066] Alvestrand, H., "Tags for the Identification of Languages", RFC 3066, BCP 47, 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

None.

1.3 Overview

The Get Rooms List Web Service protocol provides a client with a list of locations of meeting rooms within the server organization. This protocol also provides the client with the list of meeting rooms within a selected location room list. Each list of locations is an unfiltered list of names of all the locations. Each list of meeting rooms in a location list contains a list of room recipients, where an e-mail address and room display name are included for each room.

1.4 Relationship to Other Protocols

The Get Rooms List Web Service protocol uses SOAP over HTTP and SOAP over HTTPS, as shown in the following figures.

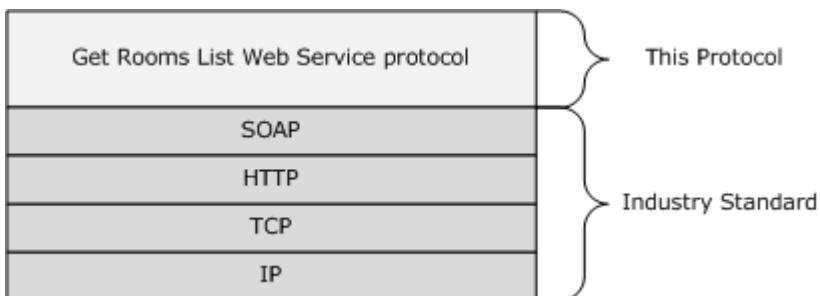


Figure 1: SOAP over HTTP

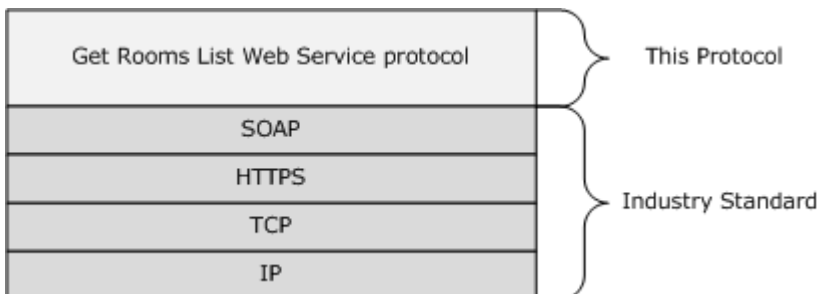


Figure 2: SOAP over HTTPS

1.5 Prerequisites/Preconditions

None.

1.6 Applicability Statement

This protocol is applicable to higher level client applications that provide information about meeting rooms for use in scheduling meetings.

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** portType version. This version is defined in section [3.1](#).
- **Security and Authentication Methods:** This protocol relies on the Web server that is hosting it to perform authentication.
- **Localization:** This protocol includes text strings in various messages. Localization considerations for such strings are specified in sections [2.2](#) and [3.1.4](#).
- **Capability Negotiation:** None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

The SOAP version supported is SOAP 1.1, as specified in [\[SOAP1.1\]](#)

2.2 Common Message Syntax

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

2.2.1 Namespaces

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

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

2.2.2 Simple Types

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

2.2.3 Complex Types

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

2.2.4 Elements

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

2.2.5 Attributes

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

2.2.6 Groups

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

2.2.7 Attribute Groups

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

2.2.8 Message Syntax

This specification does not define any common XML schema message syntax definitions.

3 Protocol Details

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

3.1 ExchangeServicePortType Server Details

The Get Rooms List Web Service protocol defines a single port type.

Operation	Description
GetRoomLists	GetRoomLists retrieves a collection of all room lists in the organization.
GetRooms	GetRooms retrieves a collection of all rooms in the specified room list in the organization.

3.1.1 Abstract Data Model

This 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 listed in the following table.

Operation	Description
GetRoomLists	Retrieves a collection of all room lists in the organization.
GetRooms	Retrieves a collection of all rooms in the specified room list in the organization.

3.1.4.1 GetRoomLists

GetRoomLists retrieves a collection of all room lists in the organization.

Request

Message Format	Description
tns:GetRoomListsSoapIn	Specifies the SOAP message that requests room lists.

Response

Message Format	Description
tns:GetRoomListsSoapOut	Specifies the SOAP message returned by the server in response.

3.1.4.1.1 GetRoomLists Complex Types

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

Type	Description
m:GetRoomListsResponseMessageType	Provides status information about the response.
m:GetRoomListsType	Provides the room lists available within the server organization.

3.1.4.1.1.1 m:GetRoomListsResponseMessageType Complex Type

The **GetRoomListsResponseMessageType** provides status information about the response.

```
<xs:complexType name="m:GetRoomListsResponseMessageType">
  <xs:complexContent>
    <xs:extension
      base="m:ResponseMessageType"
    >
      <xs:sequence>
        <xs:element name="RoomLists"
          type="t:ArrayOfEmailAddressesType"
          minOccurs="0"
        />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Child Elements

Element	Type	Description
RoomLists	t:ArrayOfEmailAddressesType	Contains the requested room list.

3.1.4.1.1.2 m:GetRoomListsType Complex Type

The **GetRoomListsType** complex type provides the room lists available within the server organization.

```
<xs:complexType name="m:GetRoomListsType">
  <xs:complexContent>
    <xs:extension
      base="m:BaseRequestType"
    />
  </xs:complexContent>
</xs:complexType>
```

3.1.4.1.2 GetRoomLists Elements

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

Operation	Description
tns:GetRoomLists	Retrieves a collection of all room lists in the organization.
tns:GetRoomListsResponse	Represents the response to a distribution list expansion operation.

3.1.4.1.2.1 GetRoomLists Element

The **GetRoomLists** element retrieves a collection of all room lists in the organization.

```
<xs:element name="GetRoomLists"
  type="m:GetRoomListsType"
/>
```

3.1.4.1.2.2 GetRoomListsResponse Element

The **GetRoomListsResponse** element represents the response to a distribution list expansion operation.

```
<xs:element name="GetRoomListsResponse"
  type="m:GetRoomListsResponseMessageType"
/>
```

3.1.4.1.3 GetRoomLists Messages

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

Messages	Description
tns:GetRoomListsSoapIn	The request that retrieves a collection of all room lists in the organization.
tns:GetRoomListsSoapOut	The response that delivers the collection of all room lists in the organization.

3.1.4.1.3.1 tns:GetRoomListsSoapIn

The [tns:GetRoomListsSoapIn](#) message contains four parts, as described in the following table.

Part Name	Element/Type	Description
GetRoomListsRequest	tns:GetRoomLists	Specifies the request.
Impersonation	t:ExchangeImpersonation	Specifies the user whom the client application is impersonating.
MailboxCulture	t:MailboxCulture	Specifies the culture to use for accessing the mailbox. The cultures are defined by [RFC3066] .
RequestVersion	t:RequestServerVersion	Specifies the schema version for the tns:GetRoomLists request.

3.1.4.1.3.2 tns:GetRoomListsSoapOut

The **tns:GetRoomListsSoapOut** message contains two parts, as described in the following table.

Part Name	Element/Type	Description
GetRoomListsResult	tns:GetRoomListsResponse	Specifies the response message.
ServerVersion	t:ServerVersionInfo	Specifies the schema version for the tns:GetRoomListsResponse message.

3.1.4.2 Get Rooms

GetRooms retrieves a collection of all rooms in the specified room list in the organization.

Request

Message Format	Description
tns:GetRoomsSoapIn	Specifies the SOAP message that requests rooms from the server.

Response

Message Format	Description
tns:GetRoomsSoapOut	Specifies the SOAP message returned by the server in response.

3.1.4.2.1 GetRooms Complex Types

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

Type	Description
m:GetRoomsResponseMessageType	Provides the status of a single <GetRoomLists> request.
m:GetRoomsType	Contains a list of rooms.
t:ArrayOfRoomsType	Contains an array of Rooms that comprise the requested room list.
t:DirectoryEntryType	Represents the item identifier for the e-mail address of a room.
t:RoomType	Contains the name and e-mail address of a room.

3.1.4.2.1.1 m:GetRoomsResponseMessageType Complex Type

The **GetRoomsResponseMessageType** complex type provides the status of a single <GetRoomLists> request.

```
<xs:complexType>
  <xs:complexContent>
    <xs:extension
      base="m:ResponseMessageType"
    >
      <xs:sequence>
```

```

    <xs:element name="Rooms"
      type="t:ArrayOfRoomsType"
      minOccurs="0"
    />
  </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

```

Child Elements

Element	Type	Description
Rooms	t:ArrayOfRoomsType	Provides an array of Rooms that comprise the requested room list.

3.1.4.2.1.2 m:GetRoomsType Complex Type

The **GetRoomsType** complex type contains the list of rooms.

```

<xs:complexType name="GetRoomsType"
  mixed="false"
>
  <xs:complexContent
    mixed="false"
  >
    <xs:extension
      base="m:BaseRequestType"
    >
      <xs:sequence>
        <xs:element name="RoomList"
          type="t:EmailAddressType"
        />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Child Elements

Element	Type	Description
RoomList	t:EmailAddressType	Contains the requested room list.

3.1.4.2.1.3 t:ArrayOfRoomsType Complex Type

The **ArrayOfRoomsType** complex type contains an array of **Rooms** that comprise the requested room list.

```

<xs:complexType name="t:ArrayOfRoomsType">
  <xs:sequence>
    <xs:element name="Room"
      type="t:RoomType"
    />
  </xs:sequence>
</xs:complexType>

```

```

    />
  </xs:sequence>
</xs:complexType>

```

Child Elements

Element	Type	Description
Room	t:RoomType	Contains the name and e-mail address of a room.

3.1.4.2.1.4 t:DirectoryEntryType Complex Type

The **DirectoryEntryType** complex type represents the item identifier for the e-mail address of a room.

```

<xs:complexType name="DirectoryEntryType">
  <xs:sequence>
    <xs:element name="Id"
      type="t:EmailAddressType"
      minOccurs="0"
    />
  </xs:sequence>
</xs:complexType>

```

Child Elements

Element	Type	Description
Id	t:EmailAddressType	Represents the item identifier for the e-mail address of a room.

3.1.4.2.1.5 t:RoomType Complex Type

The RoomType complex type contains the name and e-mail address of a room.

```

<xs:complexType name="t:RoomType">
  <xs:complexContent>
    <xs:extension
      base="t:DirectoryEntryType"
    />
  </xs:complexContent>
</xs:complexType>

```

3.1.4.2.2 GetRooms Elements

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

Operation	Description
tns:GetRooms	Retrieves all the rooms within a particular room list.

Operation	Description
tns:GetRoomsResponse	Defines a response to a GetRooms request.

3.1.4.2.2.1 tns:GetRooms Element

The **GetRooms** element retrieves all the rooms within a particular room list.

```
<xs:element name="tns:GetRooms"
  type="m:GetRoomsType"
 />
```

3.1.4.2.2.2 tns:GetRoomsResponse Element

The **GetRoomsResponse** element defines a response to the **GetRooms** request.

```
<xs:element name="tns:GetRoomsResponse"
  type="m:GetRoomsResponseMessageType"
 />
```

3.1.4.2.3 GetRooms Messages

The following WSDL message definitions are specific to this operation.

Messages	Description
tns:GetRoomsSoapIn	The request that retrieves a list of rooms from a particular room list.
tns:GetRoomsSoapOut	The response that delivers a list of rooms from a particular room list.

3.1.4.2.3.1 tns:GetRoomsSoapIn

The **GetRoomsSoapIn** message contains four parts, as described in the following table.

Part Name	Element/Type	Description
GetRoomsRequest	tns:GetRooms	Retrieves all the rooms within a particular room list.
Impersonation	t:ExchangeImpersonation	Specifies the user whom the client application is impersonating.
MailboxCulture	t:MailboxCulture	Specifies the culture to use for accessing the mailbox. The cultures are defined by [RFC3066] .
RequestVersion	t:RequestServerVersion	Specifies the schema version for the tns:GetRooms request.

3.1.4.2.3.2 tns:GetRoomsSoapOut

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

Part Name	Element/Type	Description
GetRoomsResult	tns:GetRoomsResponse	Specifies the response message.
ServerVersion	t:ServerVersionInfo	Specifies the schema version for the GetRoomsSoapOut message.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

3.2 Client Details

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

3.2.1 Client Abstract Data Model

None.

3.2.2 Client Timers

None.

3.2.3 Client Initialization

None.

3.2.4 Client Message Processing Events and Sequencing

None.

3.2.5 Client Timer Events

None.

3.2.6 Client Other Local Events

None.

4 Protocol Examples

None.

5 Security

5.1 Security Considerations for Implementors

The Get Rooms List Web service does not use additional security mechanisms.

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

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

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

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

6.1 WSDL

This section contains the contents of the MS-OXWSGTRM.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:import
namespace="http://schemas.microsoft.com/exchange/services/2006/types"/>
      <xs:include schemaLocation="MS-OXWSGTRM-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"/>
      <!-- Add global elements and types from types.xsd -->
    </xs:schema>
  </wsdl:types>
  <wsdl:portType name="ExchangeServicePortType">
    <wsdl:operation name="GetRoomLists">
      <wsdl:input message="tns:GetRoomListsSoapIn"/>
      <wsdl:output message="tns:GetRoomListsSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="GetRooms">
```

```

        <wsdl:input message="tns:GetRoomsSoapIn"/>
        <wsdl:output message="tns:GetRoomsSoapOut"/>
    </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="GetRoomLists">
        <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/GetRoomLists"/>
        <wsdl:input>
            <soap:body parts="GetRoomListsRequest" use="literal"/>
            <soap:header message="tns:GetRoomListsSoapIn" part="Impersonation"
use="literal"/>
            <soap:header message="tns:GetRoomListsSoapIn" part="MailboxCulture"
use="literal"/>
            <soap:header message="tns:GetRoomListsSoapIn" part="RequestVersion"
use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body parts="GetRoomListsResult" use="literal"/>
            <soap:header message="tns:GetRoomListsSoapOut" part="ServerVersion"
use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetRooms">
        <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/GetRooms"/>
        <wsdl:input>
            <soap:body parts="GetRoomsRequest" use="literal"/>
            <soap:header message="tns:GetRoomsSoapIn" part="Impersonation"
use="literal"/>
            <soap:header message="tns:GetRoomsSoapIn" part="MailboxCulture"
use="literal"/>
            <soap:header message="tns:GetRoomsSoapIn" part="RequestVersion"
use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body parts="GetRoomsResult" use="literal"/>
            <soap:header message="tns:GetRoomsSoapOut" part="ServerVersion"
use="literal"/>
        </wsdl:output>
    </wsdl:operation>
</wsdl:binding>
<wsdl:message name="GetRoomListsSoapIn">
    <wsdl:part name="GetRoomListsRequest" element="tns:GetRoomLists"/>
    <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="GetRoomListsSoapOut">
    <wsdl:part name="GetRoomListsResult" element="tns:GetRoomListsResponse"/>
    <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
<wsdl:message name="GetRoomsSoapIn">
    <wsdl:part name="GetRoomsRequest" element="tns:GetRooms"/>
    <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="GetRoomsSoapOut">
        <wsdl:part name="GetRoomsResult" element="tns:GetRoomsResponse"/>
        <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
    </wsdl:message>
</wsdl:definitions>

```

6.2 Types Schema

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

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

Defining specification	File name
[MS-OXWSCORE] , section 6.2	MS-OXWSCORE-types.xsd

```

<?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-OXWSCORE-types.xsd"/>
    <xs:complexType name="DirectoryEntryType">
        <xs:sequence>
            <xs:element name="Id" type="t:EmailAddressType" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="RoomType">
        <xs:complexContent>
            <xs:extension base="t:DirectoryEntryType"/>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="ArrayOfRoomsType">
        <xs:sequence>
            <xs:element name="Room" type="t:RoomType" minOccurs="0"
maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:schema>

```

6.3 Messages Schema

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

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

Defining specification	File name
[MS-OXWSCDATA] , section 6.2	MS-OXWSCDATA-messages.xsd

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
elementFormDefault="qualified" version="Exchange2010" id="messages">
  <xs:import namespace="http://schemas.microsoft.com/exchange/services/2006/types"
schemaLocation="MS-OXWSGTRM-types.xsd"/>
  <xs:include schemaLocation="MS-OXWSCDATA-messages.xsd"/>
  <xs:complexType name="GetRoomListsType">
    <xs:complexContent>
      <xs:extension base="m:BaseRequestType"/>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="GetRoomLists" type="m:GetRoomListsType"/>
  <xs:complexType name="GetRoomListsResponseMessageType">
    <xs:complexContent>
      <xs:extension base="m:ResponseMessageType">
        <xs:sequence>
          <xs:element name="RoomLists" type="t:ArrayOfEmailAddressesType"
minOccurs="0"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="GetRoomListsResponse" type="m:GetRoomListsResponseMessageType"/>
  <xs:complexType name="GetRoomsType" mixed="false">
    <xs:complexContent mixed="false">
      <xs:extension base="m:BaseRequestType">
        <xs:sequence>
          <xs:element name="RoomList" type="t:EmailAddressType"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="GetRooms" type="m:GetRoomsType"/>
  <xs:complexType name="GetRoomsResponseMessageType">
    <xs:complexContent>
      <xs:extension base="m:ResponseMessageType">
        <xs:sequence>
          <xs:element name="Rooms" type="t:ArrayOfRoomsType" minOccurs="0"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="GetRoomsResponse" type="m:GetRoomsResponseMessageType"/>
</xs:schema>
```

7 Appendix B: Product Behavior

The information in this specification is applicable to the following product versions. References to product versions include released service packs.

- Microsoft® Exchange Server 2010

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

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

8 Change Tracking

This section identifies changes made to [MS-OXWSGTRM] protocol documentation between February 2010 and May 2010 releases. Changes are classed as major, minor, or editorial.

Major changes affect protocol interoperability or implementation. Examples of major changes are:

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

Minor changes do not affect protocol interoperability or implementation. Examples are updates to fix technical accuracy or ambiguity at the sentence, paragraph, or table level.

Editorial changes apply to grammatical, formatting, and style issues.

No changes means that the document is identical to its last release.

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

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

- Content removed for template compliance.
- Obsolete document removed.

Editorial changes always have the revision type "Editorially updated."

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

Protocol syntax refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.

Protocol revision refers to changes made to a protocol that affect the bits that are sent over the wire.

Changes are listed in the following table. If you need further information, please contact protocol@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Revision Type
1.3 Overview	Updated the section title.	N	Content updated for template compliance.

9 Index

A

[Applicability](#) 7

C

[Capability negotiation](#) 7

[Change tracking](#) 25

Client

[overview](#) 17

F

[Full WSDL](#) 20

G

[Glossary](#) 5

I

[Introduction](#) 5

M

Messages

[syntax](#) 8

[transport](#) 8

O

[Overview](#) 6

P

[Product behavior](#) 24

R

References

[informative](#) 6

[normative](#) 5

[Relationship to other protocols](#) 6

S

Security

implementer considerations ([section 1.5](#) 7,
[section 5.1](#) 19)

[parameter index](#) 19

Server

[overview](#) 10

[Standards assignments](#) 7

T

[Tracking changes](#) 25

V

[Vendor-extensible fields](#) 7

[Versioning](#) 7