

[MS-MEETS]:

Meetings Web Services Protocol

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 [Open Specification Promise](#) or the [Community Promise](#). 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. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.
- **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
4/4/2008	0.1	New	Initial Availability
6/27/2008	1.0	Major	Revised and edited the technical content
12/12/2008	1.01	Editorial	Revised and edited the technical content
7/13/2009	1.02	Major	Revised and edited the technical content
8/28/2009	1.03	Editorial	Revised and edited the technical content
11/6/2009	1.04	Editorial	Revised and edited the technical content
2/19/2010	2.0	Editorial	Revised and edited the technical content
3/31/2010	2.01	Editorial	Revised and edited the technical content
4/30/2010	2.02	Editorial	Revised and edited the technical content
6/7/2010	2.03	Editorial	Revised and edited the technical content
6/29/2010	2.04	Minor	Clarified the meaning of the technical content.
7/23/2010	2.05	Editorial	Changed language and formatting in the technical content.
9/27/2010	2.05	None	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	2.05	None	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	2.06	Minor	Clarified the meaning of the technical content.
3/18/2011	2.06	None	No changes to the meaning, language, or formatting of the technical content.
6/10/2011	2.06	None	No changes to the meaning, language, or formatting of the technical content.
1/20/2012	3.0	Major	Significantly changed the technical content.
4/11/2012	3.0	None	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	3.1	Minor	Clarified the meaning of the technical content.
9/12/2012	3.1	None	No changes to the meaning, language, or formatting of the technical content.
10/8/2012	3.2	Minor	Clarified the meaning of the technical content.
2/11/2013	4.0	Major	Significantly changed the technical content.
7/30/2013	4.0	None	No changes to the meaning, language, or formatting of the technical content.
11/18/2013	4.0	None	No changes to the meaning, language, or formatting of the technical content.
2/10/2014	4.0	None	No changes to the meaning, language, or formatting of the technical content.

Date	Revision History	Revision Class	Comments
4/30/2014	4.1	Minor	Clarified the meaning of the technical content.
7/31/2014	4.1	None	No changes to the meaning, language, or formatting of the technical content.
10/30/2014	4.1	None	No changes to the meaning, language, or formatting of the technical content.
3/16/2015	5.0	Major	Significantly changed the technical content.
2/26/2016	6.0	Major	Significantly changed the technical content.

Table of Contents

1	Introduction	7
1.1	Glossary	7
1.2	References	8
1.2.1	Normative References	9
1.2.2	Informative References	9
1.3	Protocol Overview (Synopsis)	9
1.4	Relationship to Other Protocols	10
1.5	Prerequisites/Preconditions	10
1.6	Applicability Statement	10
1.7	Versioning and Capability Negotiation	10
1.8	Vendor-Extensible Fields	11
1.9	Standards Assignments.....	11
2	Messages.....	12
2.1	Transport	12
2.2	Common Message Syntax	12
2.2.1	Namespaces	12
2.2.2	Messages.....	12
2.2.3	Elements	12
2.2.3.1	Detail	13
2.2.3.2	IcalText	13
2.2.3.3	Lcid.....	13
2.2.3.4	RecurrenceId.....	13
2.2.3.5	Sequence.....	13
2.2.3.6	UID.....	13
2.2.4	Complex Types.....	14
2.2.4.1	AddMeeting	14
2.2.4.2	AttendeeUpdateStatus	14
2.2.5	Simple Types	15
2.2.5.1	CaseInsensitiveTrueFalse	15
2.2.5.2	CaseInsensitiveTrueFalseOrEmpty	15
2.2.5.3	IntOrEmpty	15
2.2.5.4	Empty	15
2.2.6	Attributes	16
2.2.7	Groups	16
2.2.8	Attribute Groups.....	16
3	Protocol Details	17
3.1	Meetings Soap Server Details	17
3.1.1	Abstract Data Model.....	17
3.1.2	Timers	17
3.1.3	Initialization.....	17
3.1.4	Message Processing Events and Sequencing Rules	17
3.1.4.1	AddMeeting	18
3.1.4.1.1	Messages	18
3.1.4.1.1.1	AddMeetingSoapIn	18
3.1.4.1.1.2	AddMeetingSoapOut.....	19
3.1.4.1.2	Elements.....	19
3.1.4.1.2.1	AddMeeting.....	19
3.1.4.1.2.2	AddMeetingResponse.....	19
3.1.4.2	AddMeetingFromICal	20
3.1.4.2.1	Messages	20
3.1.4.2.1.1	AddMeetingFromICalSoapIn	20
3.1.4.2.1.2	AddMeetingFromICalSoapOut	21
3.1.4.2.2	Elements.....	21

3.1.4.2.2.1	AddMeetingFromICal	21
3.1.4.2.2.2	AddMeetingFromICalResponse	21
3.1.4.2.3	Complex Types	22
3.1.4.2.3.1	AddMeetingFromICal	22
3.1.4.3	CreateWorkspace	22
3.1.4.3.1	Messages	22
3.1.4.3.1.1	CreateWorkspaceSoapIn	22
3.1.4.3.1.2	CreateWorkspaceSoapOut	22
3.1.4.3.2	Elements	23
3.1.4.3.2.1	CreateWorkspace	23
3.1.4.3.2.2	CreateWorkspaceResponse	23
3.1.4.3.3	Complex Types	24
3.1.4.3.3.1	TimeZoneInf	24
3.1.4.3.3.2	SysTime	24
3.1.4.4	DeleteWorkspace	25
3.1.4.4.1	Messages	25
3.1.4.4.1.1	DeleteWorkspaceSoapIn	25
3.1.4.4.1.2	DeleteWorkspaceSoapOut	26
3.1.4.4.2	Elements	26
3.1.4.4.2.1	DeleteWorkspace	26
3.1.4.4.2.2	DeleteWorkspaceResponse	26
3.1.4.5	GetMeetingsInformation	26
3.1.4.5.1	Messages	26
3.1.4.5.1.1	GetMeetingsInformationSoapIn	26
3.1.4.5.1.2	GetMeetingsInformationSoapOut	27
3.1.4.5.2	Elements	27
3.1.4.5.2.1	GetMeetingsInformation	27
3.1.4.5.2.2	GetMeetingsInformationResponse	27
3.1.4.6	GetMeetingWorkspaces	30
3.1.4.6.1	Messages	30
3.1.4.6.1.1	GetMeetingWorkspacesSoapIn	30
3.1.4.6.1.2	GetMeetingWorkspacesSoapOut	30
3.1.4.6.2	Elements	30
3.1.4.6.2.1	GetMeetingWorkspaces	30
3.1.4.6.2.2	GetMeetingWorkspacesResponse	30
3.1.4.7	RemoveMeeting	31
3.1.4.7.1	Messages	31
3.1.4.7.1.1	RemoveMeetingSoapIn	31
3.1.4.7.1.2	RemoveMeetingSoapOut	32
3.1.4.7.2	Elements	32
3.1.4.7.2.1	RemoveMeeting	32
3.1.4.7.2.2	RemoveMeetingResponse	32
3.1.4.8	RestoreMeeting	33
3.1.4.8.1	Messages	33
3.1.4.8.1.1	RestoreMeetingSoapIn	33
3.1.4.8.1.2	RestoreMeetingSoapOut	33
3.1.4.8.2	Elements	33
3.1.4.8.2.1	RestoreMeeting	33
3.1.4.8.2.2	RestoreMeetingResponse	34
3.1.4.9	SetAttendeeResponse	34
3.1.4.9.1	Messages	34
3.1.4.9.1.1	SetAttendeeResponseSoapIn	34
3.1.4.9.1.2	SetAttendeeResponseSoapOut	34
3.1.4.9.2	Elements	34
3.1.4.9.2.1	SetAttendeeResponse	34
3.1.4.9.2.2	SetAttendeeResponseResponse	35
3.1.4.9.3	Simple Types	35
3.1.4.9.3.1	AttendeeResponse	35

3.1.4.10	SetWorkspaceTitle	36
3.1.4.10.1	Messages	36
3.1.4.10.1.1	SetWorkspaceTitleSoapIn	36
3.1.4.10.1.2	SetWorkspaceTitleSoapOut	36
3.1.4.10.2	Elements	36
3.1.4.10.2.1	SetWorkspaceTitle	36
3.1.4.10.2.2	SetWorkspaceTitleResponse	37
3.1.4.11	UpdateMeeting	37
3.1.4.11.1	Messages	37
3.1.4.11.1.1	UpdateMeetingSoapIn	37
3.1.4.11.1.2	UpdateMeetingSoapOut	38
3.1.4.11.2	Elements	38
3.1.4.11.2.1	UpdateMeeting	38
3.1.4.11.2.2	UpdateMeetingResponse	38
3.1.4.12	UpdateMeetingFromICal	39
3.1.4.12.1	Messages	39
3.1.4.12.1.1	UpdateMeetingFromICalSoapIn	39
3.1.4.12.1.2	UpdateMeetingFromICalSoapOut	39
3.1.4.12.2	Elements	39
3.1.4.12.2.1	UpdateMeetingFromICal	39
3.1.4.12.2.2	UpdateMeetingFromICalResponse	40
3.1.5	Timer Events	40
3.1.6	Other Local Events	40
4	Protocol Examples	41
4.1	Create a New Meeting Workspace and Manage Meetings	41
4.1.1	Find Existing Meeting Workspaces	41
4.1.2	Query for Meeting Workspace Site Templates	42
4.1.3	Create a New Meeting Workspace	43
4.1.4	Add a New Meeting	44
4.1.5	Update the Meeting	45
4.1.6	Respond to the Meeting	45
4.1.7	Remove the Meeting	46
4.1.8	Restore the Meeting	47
5	Security	48
5.1	Security Considerations for Implementers	48
5.2	Index of Security Parameters	48
6	Appendix A: Full WSDL	49
7	Appendix B: Product Behavior	60
8	Change Tracking	61
9	Index	63

1 Introduction

The Meetings Web Services Protocol enables a protocol client to create and update meeting workspaces.

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

1.1 Glossary

This document uses the following terms:

absolute URL: The full Internet address of a page or other World Wide Web resource. The absolute URL includes a protocol, such as "http," a network location, and an optional path and file name — for example, <http://www.treyresearch.net/>.

Coordinated Universal Time (UTC): A high-precision atomic time standard that approximately tracks Universal Time (UT). It is the basis for legal, civil time all over the Earth. Time zones around the world are expressed as positive and negative offsets from UTC. In this role, it is also referred to as Zulu time (Z) and Greenwich Mean Time (GMT). In these specifications, all references to UTC refer to the time at UTC-0 (or GMT).

delegate: A user or resource that has permissions to act on behalf of another user or resource.

delegator: A user or resource for which another user or resource has permission to act on its behalf.

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

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

iCalendar: A standard content type for capturing calendar event information, as described in [\[RFC2445\]](#).

language code identifier (LCID): A 32-bit number that identifies the user interface human language dialect or variation that is supported by an application or a client computer.

meeting instance: A collection of data for a meeting that occurs only once or a single occurrence of a meeting that occurs multiple times. The data can be stored in a client application or on a website.

Meeting Workspace site: A SharePoint site that is based on a Meeting Workspace site template and has a template ID value of "2". A Meeting Workspace site is used for planning, posting, and working together on meeting materials.

parent site: The site that is above the current site in the hierarchy of the site collection.

site template: An XML-based definition of site settings, including formatting, lists, views, and elements such as text, graphics, page layout, and styles. Site templates are stored in .stp files in the content database.

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

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

SOAP fault: A container for error and status information within a SOAP message. See [\[SOAP1.2-1/2007\]](#) section 5.4 for more information.

SOAP fault code: The algorithmic mechanism for identifying a **SOAP fault**. See [\[SOAP1.2-1/2007\]](#) section 5.6 for more information.

SOAP fault detail: A string containing a human-readable explanation of a **SOAP fault**, which is not intended for algorithmic processing. See [\[SOAP1.2-1/2007\]](#) section 5.4.5 for more information.

SOAP operation: An action that can be performed by a Simple Object Access Protocol (SOAP) service, as described in [\[SOAP1.1\]](#).

subsite: A complete website that is stored in a named subdirectory of another website. The parent website can be the top-level site of a site collection or another subsite. Also referred to as subweb.

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

unique identifier (UID): A pair consisting of a GUID and a version sequence number to identify each resource uniquely. The UID is used to track the object for its entire lifetime through any number of times that the object is modified or renamed.

Web Services Description Language (WSDL): An XML format for describing network services as a set of endpoints that operate on messages that contain either document-oriented or procedure-oriented information. The operations and messages are described abstractly and are bound to a concrete network protocol and message format in order to define an endpoint. Related concrete endpoints are combined into abstract endpoints, which describe a network service. WSDL is extensible, which allows the description of endpoints and their messages regardless of the message formats or network protocols that are used.

website: A group of related pages and data within a SharePoint site collection. The structure and content of a site is based on a site definition. Also referred to as SharePoint site and site.

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

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

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

1.2 References

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

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information.

[MS-DTYP] Microsoft Corporation, "[Windows Data Types](#)".

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

[RFC2445] Dawson, F., and Stenerson, D., "Internet Calendaring and Scheduling Core Object Specification (iCalendar)", RFC 2445, November 1998, <http://www.rfc-editor.org/rfc/rfc2445.txt>

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

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

[SOAP1.2/1] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H.F., "SOAP Version 1.2 Part 1: Messaging Framework", W3C Recommendation, June 2003, <http://www.w3.org/TR/2003/REC-soap12-part1-20030624>

[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., et al., Eds., "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation, December 2009, <http://www.w3.org/TR/2009/REC-xml-names-20091208/>

[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.V., Ed. and Malhotra, A., Ed., "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 Protocol Overview (Synopsis)

This protocol provides operations to create and update meeting workspaces.

A **Meeting Workspace site** is a convenient, centralized place for project collaboration and meeting proceedings. For example, after a meeting, results are published on the Meeting Workspace site. The most current versions of team documents reside on the workspace site, so attendees can send fewer files via e-mail when content is changed.

Each method in the protocol is a **SOAP operation** that accepts a set of parameters as a SOAP request and returns a set of values as a SOAP response. The protocol client sends a request to the protocol server in a **SOAP request** message, and the protocol server sends return values to the protocol client in a **SOAP response** message.

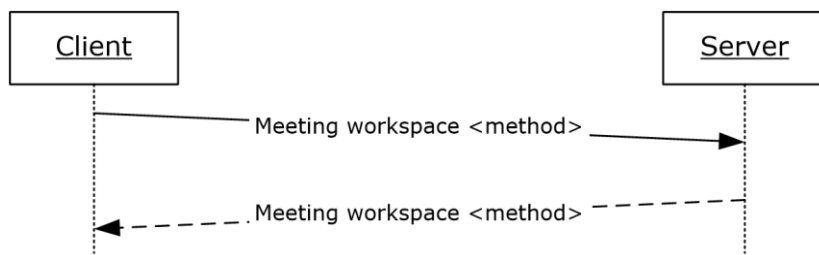


Figure 1: Client-server request and response

1.4 Relationship to Other Protocols

This protocol uses the SOAP message protocol for formatting request and response messages, as described in [SOAP1.1], [SOAP1.2/1] and [SOAP1.2/2]. It transmits those messages by using **HTTP**, as described in [RFC2616], or **Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)**, as described in [RFC2818].

The following diagram shows the underlying messaging and transport stack used by the protocol:

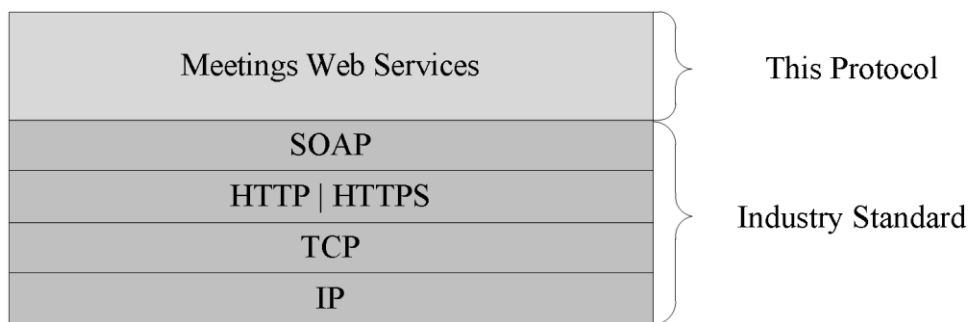


Figure 2: This protocol in relation to other protocols

The protocol has no interactions with parallel protocols. There are no other protocols that substitute for it.

1.5 Prerequisites/Preconditions

This protocol operates against a **Web site** that is identified by a **Uniform Resource Locator (URL)** that is known by protocol clients. The protocol server endpoint is formed by appending "/_vti_bin/meetings.asmx" to the URL of the Web site—for example, www.contoso.com/Repository/_vti_bin/meetings.asmx.

This protocol assumes that authentication has been performed by the underlying protocols.

1.6 Applicability Statement

This protocol can be used when there are requirements to communicate meeting data (attendees, meeting time, duration, subject, location, and other metadata) to a remote protocol server.

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

- **Supported Transports:** This protocol uses multiple transports with SOAP as specified in section [2.1](#).

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

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

2.1 Transport

Protocol servers **MUST** support SOAP over HTTP. Protocol servers **SHOULD** additionally support SOAP over HTTPS for securing communication with protocol clients.

Protocol messages **MUST** be formatted as specified in [\[SOAP1.1\]](#) section 4, SOAP Envelope, or in [\[SOAP1.2/1\]](#) section 5, SOAP Message Construct. Protocol server faults **MUST** be returned by using HTTP status codes as specified in [\[RFC2616\]](#) section 10, Status Code Definitions, or using **SOAP faults** as specified in [\[SOAP1.1\]](#) section 4.4, SOAP fault or [\[SOAP1.2/1\]](#) section 5.4, SOAP fault.

2.2 Common Message Syntax

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

2.2.1 Namespaces

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

The following table lists these namespaces, their prefixes, and the reference in which they are specified.

Prefix	Namespace URI	Reference
s	www.w3.org/2001/XMLSchema	[XMLSCHEMA1]
soap	schemas.xmlsoap.org/wsdl/soap/	[WSDL]
soap12	schemas.xmlsoap.org/wsdl/soap12/	[WSDL]
tns	schemas.microsoft.com/sharepoint/soap/meetings/	
wsdl	schemas.xmlsoap.org/wsdl/	[WSDL]

2.2.2 Messages

This specification does not define any common WSDL message definitions.

2.2.3 Elements

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

Element	Description
detail	Error information associated with a SOAP fault.
icalText	Information about a meeting instance .
lcid	A language code identifier (LCID) for a workspace.
recurrenceId	An identifier that represents an instance of a recurring meeting.
sequence	A sequence or revision number of a meeting instance.
UID	A unique identifier (UID) that represents a meeting.

2.2.3.1 Detail

The **Detail** element contains error information for a SOAP fault. It consists of a **SOAP fault code** combined with **SOAP fault detail** text that describes the error. Whenever an operation in this protocol fails, it returns a SOAP fault in this format. This element is defined as follows.

```
<s:element name="detail">
  <s:complexType>
    <s:sequence>
      <s:element name="errorstring" type="s:string" minOccurs="1" maxOccurs="1"/>
      <s:element name="errorcode" type="s:string" minOccurs="0" maxOccurs="1"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

2.2.3.2 IcalText

The **IcalText** element contains information about a meeting instance. This MUST be a valid **iCalendar** object, as specified in [\[RFC2445\]](#).

2.2.3.3 Lcid

The **Lcid** element contains a LCID.

2.2.3.4 RecurrenceId

The **RecurrenceId** element contains an identifier that specifies an instance of a recurring meeting, as specified in [\[RFC2445\]](#).

2.2.3.5 Sequence

The **Sequence** element contains a sequence or revision number of a meeting instance, as specified in [\[RFC2445\]](#).

2.2.3.6 UID

The **UID** element contains an unique identifier specifying a meeting. This MUST be a valid UID string, as used by calendaring objects and specified in [\[RFC2445\]](#). For a recurring meeting, each meeting instance shares a common **UID**. To distinguish between these meeting instances, each recurring meeting instance is also assigned a **RecurrenceID** element (section [2.2.3.4](#)). This is the date on which that meeting instance occurs in the series that was calculated with the time zone of the meeting organizer.

2.2.4 Complex Types

The following table summarizes the set of common XML schema complex type definitions used by this protocol. XML schema complex types that are specific to a particular operation are specified with the operation in section [3](#).

Complex type	Description
AddMeeting	Information about a meeting instance newly added to a workspace.
AttendeeUpdateStatus	Information about a meeting instance newly added or updated in a workspace.

2.2.4.1 AddMeeting

The **AddMeeting** complex type specifies information about a meeting instance newly added to a workspace. This type is defined as follows.

```
<s:complexType name="AddMeeting">
  <s:attribute name="Url" type="s:string"/>
  <s:attribute name="HostTitle" type="s:string"/>
  <s:attribute name="UniquePermissions" type="s:boolean"/>
  <s:attribute name="MeetingCount" type="s:int"/>
  <s:attribute name="AnonymousAccess" type="s:boolean"/>
  <s:attribute name="AllowAuthenticatedUsers" type="s:boolean"/>
</s:complexType>
```

Url: The **absolute URL** of the meeting instance in the workspace, with an indicator of the instance in the absolute URL query section.

HostTitle: The title of the site in which the meeting workspace is located.

UniquePermissions: Specifies whether the meeting workspace **subsite** contains unique role assignments.

MeetingCount: The number of meeting instances in the workspace, including the one just added.

AnonymousAccess: Specifies whether anonymous access has been granted for a meeting workspace subsite.

AllowAuthenticatedUsers: Specifies whether access to a meeting workspace subsite has been granted to authenticated users.

2.2.4.2 AttendeeUpdateStatus

The **AttendeeUpdateStatus** complex type specifies information about a meeting instance newly added or updated in a workspace. This type is defined as follows.

```

<s:complexType name="AttendeeUpdateStatus">
  <s:attribute name="Code" type="s:int"/>
  <s:attribute name="Detail" type="s:string"/>
  <s:attribute name="ManageUserPage" type="s:string"/>
</s:complexType>

```

Code: An error code associated with the text message that describes the problem that the protocol server encountered while updating the meeting instance. This number is set to zero, if there was no error.

Detail: The error that occurred during the update of the meeting instance. This string is empty, if there was no error.

ManageUserPage: The URL of a page in the meeting workspace that displays the list of attendees in the meeting instance for viewing or editing.

2.2.5 Simple Types

The following describes the XML schema simple type definitions defined in this specification. XML schema simple type definitions that are specific to a particular operation are described with the operation.

2.2.5.1 CaseInsensitiveTrueFalse

The definition of the **CaseInsensitiveTrueFalse** simple type is as follows.

```

<s:simpleType name="CaseInsensitiveTrueFalse">
  <s:restriction base="s:string">
    <s:pattern value="[Tt][Rr][Uu][Ee]|[Ff][Aa][Ll][Ss][Ee]"/>
  </s:restriction>
</s:simpleType>

```

2.2.5.2 CaseInsensitiveTrueFalseOrEmpty

The definition of the **CaseInsensitiveTrueFalseOrEmpty** simple type is as follows.

```

<s:simpleType name="CaseInsensitiveTrueFalseOrEmpty">
  <s:restriction base="s:string">
    <s:pattern value="([Tt][Rr][Uu][Ee]|[Ff][Aa][Ll][Ss][Ee])?" />
  </s:restriction>
</s:simpleType>

```

2.2.5.3 IntOrEmpty

The definition of the **IntOrEmpty** simple type is as follows.

```

<s:simpleType name="IntOrEmpty">
  <s:union memberTypes="s:int tns:Empty"/>
</s:simpleType>

```

The **Empty** simple type is specified in section [2.2.5.4](#).

2.2.5.4 Empty

The definition of the **Empty** simple type is as follows.

```
<s:simpleType name="Empty">
  <s:restriction base="s:string">
    <s:maxLength value="0"/>
  </s:restriction>
</s:simpleType>
```

2.2.6 Attributes

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

2.2.7 Groups

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

2.2.8 Attribute Groups

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

3 Protocol Details

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

The protocol client side of this protocol is a pass-through. Neither additional timers nor other states are required from the protocol client. 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 Meetings Soap Server Details

3.1.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this document.

This protocol enables a Web server to create and update meeting workspaces. These workspaces are for online collaboration. A user can access the Web site managed by the protocol to attend a meeting or a series of meetings, each of which has its own meeting workspace.

Data type	Description
Web site	Used when creating a new meeting workspace subsite.
Meeting workspace	Each Web site has zero or more meeting workspace subsites.
Meeting instance	Each meeting workspace subsite contains zero or more meeting instances.
Attendee	Each meeting instance in a meeting workspace contains one or more attendees.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing Rules

The following table summarizes the list of operations as defined by this specification.

Operation	Description
AddMeeting	Adds a meeting to an existing workspace.

Operation	Description
AddMeetingFromICal	Adds a meeting to an existing workspace, based on a iCalendar object.
CreateWorkspace	Creates a new workspace subsite.
DeleteWorkspace	Deletes a workspace.
GetMeetingsInformation	Queries information from a Web site about meeting workspaces on it or information about a meeting workspace itself.
GetMeetingWorkspaces	Obtains a list of existing meeting workspace subsites from a parent site .
RemoveMeeting	Deletes a meeting from an existing workspace.
RestoreMeeting	Restores a previously deleted meeting to a workspace.
SetAttendeeResponse	Specifies attendee response to a meeting request in a workspace.
SetWorkspaceTitle	Specifies a user-friendly name for a workspace.
UpdateMeeting	Updates content of a meeting in a workspace.
UpdateMeetingFromICal	Updates content of a meeting in a workspace, based on a iCalendar object.

3.1.4.1 AddMeeting

The **AddMeeting** operation adds a meeting to an existing workspace. This operation is defined as follows.

```
<wsdl:operation name="AddMeeting">
  <wsdl:input message="AddMeetingSoapIn" />
  <wsdl:output message="AddMeetingSoapOut" />
</wsdl:operation>
```

The protocol client sends an **AddMeetingSoapIn** request message (section [3.1.4.1.1.1](#)), and the protocol server responds with an **AddMeetingSoapOut** response message (section [3.1.4.1.1.2](#)).

This operation MUST be used when adding meetings not based on the Gregorian calendar, because the iCalendar format as used in the **AddMeetingFromICal** operation (section [3.1.4.2](#)) does not support non-Gregorian recurring meetings. An implementation SHOULD use **AddMeetingFromICal** in all other instances.

This operation MUST NOT be used to add meetings to a recurring meeting workspace.

3.1.4.1.1 Messages

3.1.4.1.1.1 AddMeetingSoapIn

The **AddMeetingSoapIn** message contains the request for the **AddMeeting** operation (section [3.1.4.1](#)).

The **SOAP action** value of the message is defined as follows.

```
http://schemas.microsoft.com/sharepoint/soap/meetings/AddMeeting
```

The **SOAP body** contains an **AddMeeting** element (section [3.1.4.1.2.1](#)).

3.1.4.1.1.2 AddMeetingSoapOut

The **AddMeetingSoapOut** message contains the response for the **AddMeeting** operation (section [3.1.4.1](#)).

The SOAP body contains an **AddMeetingResponse** element (section [3.1.4.1.2.2](#)).

3.1.4.1.2 Elements

3.1.4.1.2.1 AddMeeting

The **AddMeeting** element specifies the input parameters for the **AddMeeting** operation (section [3.1.4.1](#)). For information about meeting instance parameters, see the common **AddMeeting** type described in section [2.2.4.1](#).

```
<s:element name="AddMeeting">
  <s:complexType>
    <s:sequence>
      <s:element name="organizerEmail" type="s:string" minOccurs="0"/>
      <s:element name="uid" type="s:string"/>
      <s:element name="sequence" type="s:unsignedInt" minOccurs="0"/>
      <s:element name="utcDateStamp" type="s:dateTime" minOccurs="0"/>
      <s:element name="title" type="s:string" minOccurs="0"/>
      <s:element name="location" type="s:string" minOccurs="0"/>
      <s:element name="utcDateStart" type="s:dateTime"/>
      <s:element name="utcDateEnd" type="s:dateTime"/>
      <s:element name="nonGregorian" type="s:boolean" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

organizerEmail: The e-mail address of the meeting organizer. This is usually a fully qualified e-mail address in the form user@domain, but it can be a simple unqualified user name if both the protocol client and the protocol server are in a default domain. If this parameter is an invalid e-mail address, the response MUST be a SOAP fault with SOAP fault code "0x0000000a".

uid: The **UID** that represents this meeting. If one meeting with this **UID** exists, the operation will update the content of this meeting just as **UpdateMeeting** operation (section [3.1.4.11](#)) does.

sequence: The sequence or revision number of this meeting instance. If this element is absent, the starting sequence number is zero.

utcDateStamp: The time at which this meeting instance was created.

title: The subject of this meeting. This string has a maximum length of 255 characters.

location: The physical or virtual location in which this meeting is to take place. This string has a maximum length of 255 characters.

utcDateStart: The time at which this meeting begins.

utcDateEnd: The time at which this meeting ends. The value MUST be greater than or equal to **utcDateStart**.

nonGregorian: Specifies whether the meeting organizer is using the Gregorian calendar. The default value is FALSE.

3.1.4.1.2.2 AddMeetingResponse

The **AddMeetingResponse** element specifies the output of **AddMeeting** operation (section [3.1.4.1](#)). This element is defined as follows.

```
<s:element name="AddMeetingResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="AddMeetingResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="AddMeeting" type="tns:AddMeeting"/>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

AddMeeting: This element contains information about a meeting instance newly added to a workspace, as specified in section [2.2.4.1](#).

If this operation is sent to a Web site that is not a meeting workspace, the response MUST be a SOAP fault with SOAP fault code "0x00000006".

If adding the meeting to a meeting workspace fails for a reason not covered by the preceding error, the response can be a SOAP fault with SOAP fault code "0x00000007" or with any other implementation-specific SOAP fault codes.

3.1.4.2 AddMeetingFromICal

The **AddMeetingFromICal** operation adds a meeting to an existing workspace, based on a iCalendar object. This operation is defined as follows.

```
<wsdl:operation name="AddMeetingFromICal">
  <wsdl:input message="AddMeetingFromICalSoapIn" />
  <wsdl:output message="AddMeetingFromICalSoapOut" />
</wsdl:operation>
```

The protocol client sends an **AddMeetingFromICalSoapIn** request message (section [3.1.4.2.1.1](#)), and the protocol server responds with an **AddMeetingFromICalSoapOut** response message (section [3.1.4.2.1.2](#)).

This operation is used to add meetings. The **AddMeeting** operation (section [3.1.4.1](#)) MUST be used when adding meetings not based on the Gregorian calendar because this operation does not support non-Gregorian recurring meetings.

3.1.4.2.1 Messages

3.1.4.2.1.1 AddMeetingFromICalSoapIn

The **AddMeetingFromICalSoapIn** message contains the request for the **AddMeetingFromICal** operation (section [3.1.4.2](#)).

The SOAP action value of the message is defined as follows.

```
http://schemas.microsoft.com/sharepoint/soap/meetings/AddMeetingFromICal
```

The SOAP body contains an **AddMeetingFromICal** element (section [3.1.4.2.2.1](#)).

3.1.4.2.1.2 AddMeetingFromICalSoapOut

The **AddMeetingFromICalSoapOut** message contains the response for the **AddMeetingFromICal** operation (section [3.1.4.2](#)).

The SOAP body contains an **AddMeetingFromICalResponse** element (section [3.1.4.2.2.2](#)).

3.1.4.2.2 Elements

3.1.4.2.2.1 AddMeetingFromICal

The **AddMeetingFromICal** element specifies the input parameters for the **AddMeetingFromICal** operation (section [3.1.4.2](#)). This operation is defined as follows.

```
<s:element name="AddMeetingFromICal">
  <s:complexType>
    <s:sequence>
      <s:element name="organizerEmail" type="s:string" minOccurs="0"/>
      <s:element name="icalText" type="s:string"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

organizerEmail: Represents the e-mail address of the meeting organizer. This is the **delegate** who sent out the meeting request. Otherwise, this element MUST be empty. When a delegate sends the meeting request, the **icalText** element represents the **delegator** on whose behalf the meeting request was sent. If this parameter is an invalid e-mail address, the response MUST be a SOAP fault with SOAP fault code "0x0000000a".

icalText: Information about the meeting instance to add. This MUST be a valid iCalendar object as specified in [\[RFC2445\]](#). The total number of attendee elements in this parameter MUST be less than or equal to 254. If this parameter contains more than 254 attendee elements, the response MUST be a SOAP fault with SOAP fault code "0x0000000d".

3.1.4.2.2.2 AddMeetingFromICalResponse

The **AddMeetingFromICalResponse** element specifies the output of the **AddMeetingFromICal** operation (section [3.1.4.2](#)). This element is defined as follows.

```
<s:element name="AddMeetingFromICalResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="AddMeetingFromICalResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="AddMeetingFromICal" type="tns:AddMeetingFromICal" />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

AddMeetingFromICalResult: The response XML consists of two elements containing information about the meeting instance newly-added to the meeting workspace.

If this operation is sent to a Web site that is not a meeting workspace, the response MUST be a SOAP fault with SOAP fault code "0x00000006".

If the protocol client tries to add a recurring meeting to a workspace that already contains a meeting, the response MUST be a SOAP fault with SOAP fault code "0x00000003".

If adding the meeting to a meeting workspace fails for a reason not covered by any of the preceding errors, the response MUST be a SOAP fault with SOAP fault code "0x00000007".

3.1.4.2.3 Complex Types

3.1.4.2.3.1 AddMeetingFromICal

The **AddMeetingFromICal** complex type specifies information about a meeting instance newly added or updated in a workspace. This type is defined as follows.

```
<s:complexType name="AddMeetingFromICal">
  <s:complexContent>
    <s:extension base="tns:AddMeeting">
      <s:sequence>
        <s:element name="AttendeeUpdateStatus" type="tns:AttendeeUpdateStatus" />
      </s:sequence>
    </s:extension>
  </s:complexContent>
</s:complexType>
```

AttendeeUpdateStatus: This element contains information about a meeting instance newly added or updated, as specified in section [2.2.4.1](#).

3.1.4.3 CreateWorkspace

The **CreateWorkspace** operation creates a new meeting workspace subsite. This operation is defined as follows.

```
<wsdl:operation name="CreateWorkspace">
  <wsdl:input message="CreateWorkspaceSoapIn" />
  <wsdl:output message="CreateWorkspaceSoapOut" />
</wsdl:operation>
```

The protocol client sends a **CreateWorkspaceSoapIn** request message (section [3.1.4.3.1.1](#)), and the protocol server responds with a **CreateWorkspaceSoapOut** response message (section [3.1.4.3.1.2](#)).

This operation is used when the protocol client is composing a meeting request message, and the user chooses to have the meeting instance linked with a meeting workspace.

3.1.4.3.1 Messages

3.1.4.3.1.1 CreateWorkspaceSoapIn

The **CreateWorkspaceSoapIn** message contains the request for the **CreateWorkspace** operation (section [3.1.4.3](#)).

The SOAP action value of the message is defined as follows.

```
http://schemas.microsoft.com/sharepoint/soap/meetings/CreateWorkspace
```

The SOAP body contains a **CreateWorkspace** element (section [3.1.4.3.2.1](#)).

3.1.4.3.1.2 CreateWorkspaceSoapOut

The **CreateWorkspaceSoapOut** message is the response message for the **CreateWorkspace** operation (section [3.1.4.3](#)).

The SOAP body contains a **CreateWorkspaceResponse** element (section [3.1.4.3.2.2](#)).

3.1.4.3.2 Elements

3.1.4.3.2.1 CreateWorkspace

The **CreateWorkspace** element specifies the input parameters for the **CreateWorkspace** operation (section [3.1.4.3](#)). This element is defined as follows.

```
<s:element name="CreateWorkspace">
  <s:complexType>
    <s:sequence>
      <s:element name="title" type="s:string" minOccurs="1"/>
      <s:element name="templateName" type="s:string" minOccurs="0"/>
      <s:element name="lcid" type="s:unsignedInt" minOccurs="0"/>
      <s:element name="timeZoneInformation" type="tns:TimeZoneInf" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

title: The title of the new meeting workspace. This string is 255 characters or less in length. Remaining characters are truncated.

templateName: The name of the **site template** to use for the new meeting workspace. "MPS" will be used if **templateName** is not set.

lcid: The LCID for the new workspace. If this parameter is absent, the LCID of the parent Web site is used. If a value for this parameter is provided, it MUST be a LCID contained in the **GetMeetingsInformationResponse** response after the client protocol sends the **GetMeetingsInformation** message (section [3.1.4.5](#)) with the **requestFlags** parameter set to 2. See section [3.1.4.5.2.1](#) and section [3.1.4.5.2.2](#) regarding the **GetMeetingsInformation** and **GetMeetingsInformationResponse** elements respectively.

timeZoneInformation: The time zone on the system of the meeting organizer.

3.1.4.3.2.2 CreateWorkspaceResponse

The **CreateWorkspaceResponse** element specifies the output of the **CreateWorkspace** operation (section [3.1.4.3](#)). This element is defined as follows.

```
<s:element name="CreateWorkspaceResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="CreateWorkspaceResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="CreateWorkspace"
              type="tns:CreateWorkspace"/>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
  <s:complexType name="CreateWorkspace">
    <s:attribute name="Url" type="s:string"/>
  </s:complexType>
```

CreateWorkspaceResult: The response XML consists of one element containing information about the newly created meeting workspace.

CreateWorkspace: This element contains information about a newly created meeting workspace.

Uri: The URL of the newly created meeting workspace.

This operation cannot create a meeting workspace as a subsite of another meeting workspace. If this operation is sent to a Web site that is a meeting workspace, the response MUST be a SOAP fault with SOAP fault code "0x00000001".

3.1.4.3.3 Complex Types

3.1.4.3.3.1 TimeZoneInf

The **TimeZoneInf** complex type specifies information about a time zone. This type is defined as follows.

```
<s:complexType name="TimeZoneInf">
  <s:sequence>
    <s:element name="bias" type="s:int"/>
    <s:element name="standardDate" type="tns:SysTime"/>
    <s:element name="standardBias" type="s:int"/>
    <s:element name="daylightDate" type="tns:SysTime"/>
    <s:element name="daylightBias" type="s:int"/>
  </s:sequence>
</s:complexType>
```

bias: The current bias for local time translation, in minutes. This bias is the difference in minutes between **Coordinated Universal Time (UTC)** and local time. Translations between UTC and local time are based on the formula: UTC = local time + bias. For example, for Eastern Standard Time, bias is 300, meaning 5 hours before UTC.

standardDate: The date and time at which standard time begins, as specified in [\[RFC2445\]](#).

standardBias: The value used during local time translations that occur during standard time. This number is added to the **bias** element to form the total bias used during standard time. In most time zones, this is zero, meaning that the clock is not adjusted during standard time.

daylightDate: The date and time at which daylight saving time begins, as specified in [\[RFC2445\]](#).

daylightBias: The value used during local time translations for daylight saving time. This number is added to the **bias** element to form the total bias used during daylight saving time. In most time zones, this is -60, meaning the clock is advanced 1 hour during daylight saving time.

3.1.4.3.3.2 SysTime

The **SysTime** complex type specifies a system date and time. The fields of this type are the same as those in the SYSTEMTIME type as specified in [\[MS-DTYP\]](#). This type is defined as follows.

```
<s:complexType name="SysTime">
  <s:sequence>
    <s:element name="year" type="s:unsignedShort"/>
    <s:element name="month" type="s:unsignedShort"/>
    <s:element name="dayOfWeek" type="s:unsignedShort"/>
    <s:element name="day" type="s:unsignedShort"/>
    <s:element name="hour" type="s:unsignedShort"/>
    <s:element name="minute" type="s:unsignedShort"/>
    <s:element name="second" type="s:unsignedShort"/>
    <s:element name="milliseconds" type="s:unsignedShort"/>
  </s:sequence>
```


</s:complexType>

year: The year. If this type is a field in the **TimeZoneInf** type and this element is zero, this element specifies that the transition to daylight saving time or standard time is a relative date that happens annually. If this element is nonzero, it specifies that the transition date is absolute and only happens once.

month: The month of the year. This number MUST be in the range from 1 through 12, where 1 is January and 12 is December.

dayOfWeek: The day of the week. This number MUST be in the range from zero through 6, where zero is Sunday and 6 is Saturday.

day: The day of the month. This number MUST be in the range from 1 through 31 and MUST be appropriate to **month** and **year** values. For example, if **month** is set to 2, or February, **day** cannot exceed 29, and cannot exceed 28 in non-leap years.

If **year** is set to zero, specifying that the transition to daylight saving time or standard time is an annual occurrence, **day** specifies the number of times this weekday occurs in the month. In this case, **day** MUST be in the range from 1 through 5, where 5 specifies the final occurrence during the month if that weekday only occurs 4 times. For example, the first Sunday in April is specified as follows: **year** is set to zero, **month** is set to 4, **dayOfWeek** is set to zero, and **day** is set to 1. The last Thursday in October is specified as follows: **year** is set to zero, **month** is set to 10, **dayOfWeek** is set to 4, and **day** is set to 5.

hour: The hour of the day. This number MUST be in the range from zero through 23, where zero is 00:00 or 12:00 am, and 23 is 23:00 or 11:00 pm.

minute: The minute of the hour. This number MUST be in the range from zero through 59.

second: The second of the minute. This number MUST be in the range from zero through 59.

milliseconds: The millisecond of the second. This number MUST be in the range from zero through 999.

3.1.4.4 DeleteWorkspace

The **DeleteWorkspace** operation deletes a workspace. This operation is defined as follows.

```
<wsdl:operation name="DeleteWorkspace">
  <wsdl:input message="DeleteWorkspaceSoapIn" />
  <wsdl:output message="DeleteWorkspaceSoapOut" />
</wsdl:operation>
```

The protocol client sends a **DeleteWorkspaceSoapIn** request message (section [3.1.4.4.1.1](#)), and the protocol server responds with a **DeleteWorkspaceSoapOut** response message (section [3.1.4.4.1.2](#)). This operation takes no parameters because it deletes the meeting workspace subsite to which the request is sent. This operation fails if sent to a Web site that is not a meeting workspace.

3.1.4.4.1 Messages

3.1.4.4.1.1 DeleteWorkspaceSoapIn

The **DeleteWorkspaceSoapIn** message contains the request for the **DeleteWorkspace** operation (section [3.1.4.4](#)).

The SOAP action value of the message is defined as follows.

<http://schemas.microsoft.com/sharepoint/soap/meetings/DeleteWorkspace>

The SOAP body contains a **DeleteWorkspace** element (section [3.1.4.4.2.1](#)).

3.1.4.4.1.2 DeleteWorkspaceSoapOut

The **DeleteWorkspaceSoapOut** message contains the response for the **DeleteWorkspace** operation (section [3.1.4.4](#)).

The SOAP body contains a **DeleteWorkspaceResponse** element (section [3.1.4.4.2.2](#)).

3.1.4.4.2 Elements

3.1.4.4.2.1 DeleteWorkspace

The **DeleteWorkspace** element specifies the input parameters for the **DeleteWorkspace** operation (section [3.1.4.4](#)). This element is defined as follows.

```
<s:element name="DeleteWorkspace">
  <s:complexType/>
</s:element>
```

3.1.4.4.2.2 DeleteWorkspaceResponse

The **DeleteWorkspaceResponse** element specifies the output of the **DeleteWorkspace** operation (section [3.1.4.4](#)). This element is defined as follows.

```
<s:element name="DeleteWorkspaceResponse">
  <s:complexType/>
</s:element>
```

If this operation is sent to a Web site that is not a meeting workspace, the response MUST be a SOAP fault with SOAP fault code "0x00000004".

3.1.4.5 GetMeetingsInformation

The **GetMeetingsInformation** operation queries information from a Web site about meeting workspaces on it or information about a meeting workspace itself. This operation is defined as follows.

```
<wsdl:operation name="GetMeetingsInformation">
  <wsdl:input message="GetMeetingsInformationSoapIn" />
  <wsdl:output message="GetMeetingsInformationSoapOut" />
</wsdl:operation>
```

The protocol client sends a **GetMeetingsInformationSoapIn** request message (section [3.1.4.5.1.1](#)), and the protocol server responds with a **GetMeetingsInformationSoapOut** response message (section [3.1.4.5.1.2](#)).

3.1.4.5.1 Messages

3.1.4.5.1.1 GetMeetingsInformationSoapIn

The **GetMeetingsInformationSoapIn** is the request message for the **GetMeetingsInformation** operation (section [3.1.4.5](#)).

The SOAP action value of the message is defined as follows.

```
http://schemas.microsoft.com/sharepoint/soap/meetings/GetMeetingsInformation
```

The SOAP body contains a **GetMeetingsInformation** element (section [3.1.4.5.2.1](#)).

3.1.4.5.1.2 GetMeetingsInformationSoapOut

The **GetMeetingsInformationSoapOut** message contains the response for the **GetMeetingsInformation** operation (section [3.1.4.5](#)).

The SOAP body contains a **GetMeetingsInformationResponse** element (section [3.1.4.5.2.2](#)).

3.1.4.5.2 Elements

3.1.4.5.2.1 GetMeetingsInformation

The **GetMeetingsInformation** element specifies the input parameters associated with the **GetMeetingsInformation** operation (section [3.1.4.5](#)). This element is defined as follows.

```
<s:element name="GetMeetingsInformation">  
  <s:complexType>  
    <s:sequence>  
      <s:element name="requestFlags" type="s:unsignedInt" minOccurs="0"/>  
      <s:element name="lcid" type="s:unsignedInt" minOccurs="0"/>  
    </s:sequence>  
  </s:complexType>  
</s:element>
```

requestFlags: This field contains bit flags that specify what information to request from the Web site. Any or all bits can be set. The following table specifies possible values for these bits.

Value	Meaning
0x1	Queries whether the user has permission to create meeting workspaces on this Web site.
0x2	Queries for the site template languages supported.
0x4	Queries for the list of site templates that are supported.
0x8	Queries other status values of a workspace.

lcid: The LCID of the meeting workspace templates to be returned. This parameter is used only when the **requestFlags** parameter has bit flag 0x4 set.

3.1.4.5.2.2 GetMeetingsInformationResponse

The **GetMeetingsInformationResponse** element specifies the output of the **GetMeetingsInformation** operation (section [3.1.4.5](#)). This element is defined as follows.

```
<s:element name="GetMeetingsInformationResponse">  
  <s:complexType>  
    <s:sequence>  
      <s:element minOccurs="0" maxOccurs="1" name="GetMeetingsInformationResult">  
        <s:complexType mixed="true">  
          <s:sequence>  
            <s:element name="MeetingsInformation">  
              <s:complexType mixed="true">
```

```

        <s:element minOccurs="0" maxOccurs="1" name="AllowCreate"
type="tns:CaseInsensitiveTrueFalse" />
        <s:element name="ListTemplateLanguages" minOccurs="0">
            <s:complexType>
                <s:sequence>
                    <s:element minOccurs="0" maxOccurs="unbounded" name="LCID"
type="s:string" />
                </s:sequence>
            </s:complexType>
        </s:element>
        <s:element name="ListTemplates" minOccurs="0">
            <s:complexType>
                <s:sequence>
                    <s:element minOccurs="0" maxOccurs="unbounded" name="Template"
type="tns:Template" />
                </s:sequence>
            </s:complexType>
        </s:element>
        <s:element minOccurs="0" maxOccurs="1" name="WorkspaceStatus"
type="tns:WorkspaceStatus" />
    </s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
<s:complexType name="Template">
    <s:attribute name="Name" type="s:string"/>
    <s:attribute name="Title" type="s:string"/>
    <s:attribute name="Id" type="s:int"/>
    <s:attribute name="Description" type="s:string"/>
    <s:attribute name="ImageUrl" type="s:string"/>
</s:complexType>
<s:complexType name="WorkspaceStatus">
    <s:attribute name="UniquePermissions" type="tns:CaseInsensitiveTrueFalseOrEmpty"/>
    <s:attribute name="MeetingCount" type="tns: IntOrEmpty"/>
    <s:attribute name="AnonymousAccess" type="tns:CaseInsensitiveTrueFalseOrEmpty"/>
    <s:attribute name="AllowAuthenticatedUsers" type="tns:CaseInsensitiveTrueFalseOrEmpty"/>
</s:complexType>

```

GetMeetingsInformationResult: The response XML consists of up to four elements, one for each **requestFlags** flag. An element is present, if its corresponding **requestFlags** was set as follows.

AllowCreate: Specifies whether the user has permission to create meeting workspaces on this subsite. This element is present in the response when bit flag 0x1 is specified in **requestFlags**. When bit flag 0x1 is specified, the operation MUST be sent to a parent Web site, as opposed to a meeting workspace subsite itself. If the operation is sent to a Web site that is a meeting workspace, the response MUST be a SOAP fault with SOAP fault code "0x00000001".

```
<s:element name="AllowCreate" type="tns:CaseInsensitiveTrueFalse" minOccurs="0"/>
```

The element type **CaseInsensitiveTrueFalse** is specified in section [2.2.5.1](#).

ListTemplateLanguages: Lists the site template languages supported. This element is present in the response when bit flag 0x2 is specified in **requestFlags**.

```

<s:element name="ListTemplateLanguages" minOccurs="0">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="unbounded" name="LCID" type="s:string"/>
        </s:sequence>
    </s:complexType>

```

```
</s:complexType>
</s:element>
```

LCID: The LCID of the available site template.

ListTemplates: The list of site templates supported. This element is present in the response when bit flag 0x4 is specified in **requestFlags**.

```
<s:element name="ListTemplates" minOccurs="0">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="unbounded" name="Template" type="tns:Template"/>
    </s:sequence>
  </s:complexType>
</s:element>
<s:complexType name="Template">
  <s:attribute name="Name" type="s:string"/>
  <s:attribute name="Title" type="s:string"/>
  <s:attribute name="Id" type="s:int"/>
  <s:attribute name="Description" type="s:string"/>
  <s:attribute name="ImageUrl" type="s:string"/>
</s:complexType>
```

Template: The available site template.

Name: The short name of the site template.

Title: The title of the site template.

Id: The identification number of the site template.

Description: A description of the site template.

ImageUrl: The URL of an image or icon of the site template.

WorkspaceStatus: The status of a workspace. This element is present in the response when bit flag 0x8 is specified in **requestFlags**.

```
<s:element name="WorkspaceStatus" type="tns:WorkspaceStatus" minOccurs="0"/>
  <s:complexType name="WorkspaceStatus">
    <s:attribute name="UniquePermissions" type="tns:CaseInsensitiveTrueFalseOrEmpty"/>
    <s:attribute name="MeetingCount" type="tns:IntOrEmpty"/>
    <s:attribute name="AnonymousAccess" type="tns:CaseInsensitiveTrueFalseOrEmpty"/>
    <s:attribute name="AllowAuthenticatedUsers" type="tns:CaseInsensitiveTrueFalseOrEmpty"/>
  </s:complexType>
</s:element>
```

UniquePermissions: Specifies whether the meeting workspace subsite contains unique role assignments. The element type **CaseInsensitiveTrueFalseOrEmpty** is specified in section [2.2.5.2](#).

MeetingCount: The number of single meeting instances on the subsite. This **MUST** be set to -1 if the meeting workspace subsite has a recurring meeting. The element type **IntOrEmpty** is specified in section [2.2.5.3](#).

AnonymousAccess: Specifies whether anonymous access has been granted for a meeting workspace subsite. The element type **CaseInsensitiveTrueFalseOrEmpty** is specified in section 2.2.5.2.

AllowAuthenticatedUsers: Specifies whether access to a meeting workspace subsite has been granted to authenticated users. The element type **CaseInsensitiveTrueFalseOrEmpty** is specified in section 2.2.5.2.

3.1.4.6 GetMeetingWorkspaces

The **GetMeetingWorkspaces** operation obtains a list of created meeting workspace subsites from a parent Web site. This operation is defined as follows.

```
<wsdl:operation name="GetMeetingWorkspaces">
  <wsdl:input message="GetMeetingWorkspacesSoapIn" />
  <wsdl:output message="GetMeetingWorkspacesSoapOut" />
</wsdl:operation>
```

The protocol client sends a **GetMeetingWorkspacesSoapIn** request message (section [3.1.4.6.1.1](#)), and the protocol server responds with a **GetMeetingWorkspacesSoapOut** response message section ([3.1.4.6.1.2](#)).

3.1.4.6.1 Messages

3.1.4.6.1.1 GetMeetingWorkspacesSoapIn

The **GetMeetingWorkspacesSoapIn** message contains the request for a list of meeting workspaces to which a meeting can be added.

The SOAP action value of the message is defined as follows.

```
http://schemas.microsoft.com/sharepoint/soap/meetings/GetMeetingWorkspaces
```

The SOAP body contains a **GetMeetingWorkspaces** element (section [3.1.4.6.2.1](#)).

3.1.4.6.1.2 GetMeetingWorkspacesSoapOut

The **GetMeetingWorkspacesSoapOut** message contains the response for the **GetMeetingWorkspaces** operation (section [3.1.4.6](#)). The protocol server MUST return only workspaces to which the protocol client can add meetings.

The SOAP body contains a **GetMeetingWorkspacesResponse** element (section [3.1.4.6.2.2](#)).

3.1.4.6.2 Elements

3.1.4.6.2.1 GetMeetingWorkspaces

The **GetMeetingWorkspaces** element specifies the input parameters for the **GetMeetingWorkspaces** operation (section [3.1.4.6](#)). This element is defined as follows.

```
<s:element name="GetMeetingWorkspaces">
  <s:complexType>
    <s:sequence>
      <s:element name="recurring" type="s:boolean" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

recurring: Specifies whether meeting workspaces returned are limited to those that can be associated with a recurring meeting. If false, empty workspaces and single instance workspaces are returned. If true, only empty workspaces are returned. If not specified, the server will treat it as false.

3.1.4.6.2.2 GetMeetingWorkspacesResponse

The **GetMeetingWorkspacesResponse** element specifies the output of **GetMeetingWorkspaces** operation (section [3.1.4.6](#)). This element is defined as follows.

```
<s:element name="GetMeetingWorkspacesResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="GetMeetingWorkspacesResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="MeetingWorkspaces" minOccurs="0">
              <s:complexType>
                <s:sequence>
                  <s:element minOccurs="0" maxOccurs="unbounded" name="Workspace"
type="tns:Workspace"/>
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
  <s:complexType name="Workspace">
    <s:attribute name="Url" type="s:string"/>
    <s:attribute name="Title" type="s:string"/>
  </s:complexType>
</s:element>
```

GetMeetingWorkspacesResult: The response XML consists of one element containing a list of meeting workspaces.

MeetingWorkspaces: A list of meeting workspaces.

Workspace: An individual meeting workspace.

Url: The URL of the meeting workspace.

Title: The title of the meeting workspace.

3.1.4.7 RemoveMeeting

The **RemoveMeeting** operation removes the association between a meeting and a meeting workspace. This operation is defined as follows.

```
<wsdl:operation name="RemoveMeeting">
  <wsdl:input message="RemoveMeetingSoapIn" />
  <wsdl:output message="RemoveMeetingSoapOut" />
</wsdl:operation>
```

The protocol client sends a **RemoveMeetingSoapIn** request message (section [3.1.4.7.1.1](#)), and the protocol server responds with a **RemoveMeetingSoapOut** response message (section [3.1.4.7.1.2](#)).

If the meeting organizer wants to remove a meeting instance from the protocol client and the meeting is associated with a meeting workspace, the protocol client calls this operation to remove the association between the meeting and the meeting workspace.

3.1.4.7.1 Messages

3.1.4.7.1.1 RemoveMeetingSoapIn

The **RemoveMeetingSoapIn** message contains the request for the **RemoveMeeting** operation (section [3.1.4.7](#)).

The SOAP action value of the message is defined as follows.

```
http://schemas.microsoft.com/sharepoint/soap/meetings/RemoveMeeting
```

The SOAP body contains a **RemoveMeeting** element (section [3.1.4.7.2.1](#)).

3.1.4.7.1.2 RemoveMeetingSoapOut

The **RemoveMeetingSoapOut** message contains the response for the **RemoveMeeting** operation (section [3.1.4.7](#)).

The SOAP body contains a **RemoveMeetingResponse** element (section [3.1.4.7.2.2](#)).

3.1.4.7.2 Elements

3.1.4.7.2.1 RemoveMeeting

The **RemoveMeeting** element specifies the input parameters for the **RemoveMeeting** operation (section [3.1.4.7](#)). This element is defined as follows.

```
<s:element name="RemoveMeeting">
  <s:complexType>
    <s:sequence>
      <s:element name="recurrenceId" type="s:unsignedInt" minOccurs="0"/>
      <s:element name="uid" type="s:string"/>
      <s:element name="sequence" type="s:unsignedInt" minOccurs="0"/>
      <s:element name="utcDateStamp" type="s:dateTime" minOccurs="0"/>
      <s:element name="cancelMeeting" type="s:boolean" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

recurrenceId: The instance of a recurring meeting to delete. For single instance meetings, this element MUST be set to zero.

uid: The **UID** specifying the meeting to delete.

sequence: The sequence number of a meeting instance to delete. The default value is zero. If the meeting was updated using the **UpdateMeeting** operation (section [3.1.4.11](#)), the sequence provided MUST be equal to or greater than the sequence input parameter for the **UpdateMeeting** element. See section [3.1.4.11.2.1](#) for more information regarding the **UpdateMeeting** element.

utcDateStamp: The time stamp for the time at which this meeting instance was deleted.

cancelMeeting: Specifies whether the meeting is being cancelled, in addition to being deleted from the workspace. If not specified, the server will treat it as false.

3.1.4.7.2.2 RemoveMeetingResponse

The **RemoveMeetingResponse** element specifies the output of the **RemoveMeeting** operation (section [3.1.4.7](#)). This element is defined as follows.

```
<s:element name="RemoveMeetingResponse">
  <s:complexType/>
```



```
</s:element>
```

3.1.4.8 RestoreMeeting

The **RestoreMeeting** operation restores a previously deleted meeting to a workspace. This operation is defined as follows.

```
<wsdl:operation name="RestoreMeeting">  
  <wsdl:input message="RestoreMeetingSoapIn" />  
  <wsdl:output message="RestoreMeetingSoapOut" />  
</wsdl:operation>
```

The protocol client sends a **RestoreMeetingSoapIn** request message (section [3.1.4.8.1.1](#)), and the protocol server responds with a **RestoreMeetingSoapOut** response message (section [3.1.4.8.1.2](#)).

This operation undoes a previous action. For example, a meeting request form on the protocol client is closed without saving, and the meeting workspace needs to be reverted to the last persisted state. This operation does not restore individual meeting instances in a recurring meeting because it does not have a *recurrenceId* parameter.

3.1.4.8.1 Messages

3.1.4.8.1.1 RestoreMeetingSoapIn

The **RestoreMeetingSoapIn** message requests that a meeting be restored.

The SOAP action value of the message is defined as follows.

```
http://schemas.microsoft.com/sharepoint/soap/meetings/RestoreMeeting
```

The SOAP body contains a **RestoreMeeting** element (section [3.1.4.8.2.1](#)).

3.1.4.8.1.2 RestoreMeetingSoapOut

The **RestoreMeetingSoapOut** response is sent when a meeting is restored.

The SOAP body contains a **RestoreMeetingResponse** element (section [3.1.4.8.2.2](#)).

3.1.4.8.2 Elements

3.1.4.8.2.1 RestoreMeeting

The **RestoreMeeting** element specifies the input parameters for the **RestoreMeeting** operation (section [3.1.4.8](#)). This element is defined as follows.

```
<s:element name="RestoreMeeting">  
  <s:complexType>  
    <s:sequence>  
      <s:element name="uid" type="s:string"/>  
    </s:sequence>  
  </s:complexType>  
</s:element>
```

uid: The **UID** specifying the meeting to restore.

3.1.4.8.2.2 RestoreMeetingResponse

The **RestoreMeetingResponse** element specifies the output of the **RestoreMeeting** operation (section [3.1.4.8](#)). This element is defined as follows.

```
<s:element name="RestoreMeetingResponse">
  <s:complexType/>
</s:element>
```

If the meeting specified by the **uid** parameter in the **RestoreMeeting** operation does not exist in the meeting workspace, a SOAP fault response is returned with SOAP fault code "0x8102003e". See section [2.2.3.1](#) regarding the SOAP fault message.

3.1.4.9 SetAttendeeResponse

The **SetAttendeeResponse** operation specifies attendee response to a meeting request in a workspace. This operation is defined as follows.

```
<wsdl:operation name="SetAttendeeResponse">
  <wsdl:input message="SetAttendeeResponseSoapIn" />
  <wsdl:output message="SetAttendeeResponseSoapOut" />
</wsdl:operation>
```

The protocol client sends a **SetAttendeeResponseSoapIn** request message (section [3.1.4.9.1.1](#)), and the protocol server responds with a **SetAttendeeResponseSoapOut** response message (section [3.1.4.9.1.2](#)).

When an attendee accepts or declines a meeting request, the protocol client checks whether the meeting is linked with a meeting workspace and, if so, updates the workspace.

3.1.4.9.1 Messages

3.1.4.9.1.1 SetAttendeeResponseSoapIn

The **SetAttendeeResponseSoapIn** message requests the **SetAttendeeResponse** operation (section [3.1.4.9](#)).

The SOAP action value of the message is defined as follows.

```
http://schemas.microsoft.com/sharepoint/soap/meetings/SetAttendeeResponse
```

The SOAP body contains a **SetAttendeeResponse** element (section [3.1.4.9.2.1](#)).

3.1.4.9.1.2 SetAttendeeResponseSoapOut

The **SetAttendeeResponseSoapOut** message contains the response to the **SetAttendeeResponse** operation (section [3.1.4.9](#)).

The SOAP body contains a **SetAttendeeResponseResponse** element (section [3.1.4.9.2.2](#)).

3.1.4.9.2 Elements

3.1.4.9.2.1 SetAttendeeResponse

The **SetAttendeeResponse** element specifies the input of **SetAttendeeResponse** operation (section [3.1.4.9](#)). This element is defined as follows.

```
<s:element name="SetAttendeeResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="attendeeEmail" type="s:string"/>
      <s:element name="recurrenceId" type="s:unsignedInt" minOccurs="0"/>
      <s:element name="uid" type="s:string"/>
      <s:element name="sequence" type="s:unsignedInt" minOccurs="0"/>
      <s:element name="utcDateTimeOrganizerCriticalChange" type="s:dateTime" minOccurs="0"/>
      <s:element name="utcDateTimeAttendeeCriticalChange" type="s:dateTime" minOccurs="0"/>
      <s:element name="response" type="tns:AttendeeResponse" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

attendeeEmail: The e-mail address of the responding attendee.

recurrenceId: The instance of a recurring meeting associated with the response. For single-instance meetings, this **MUST** be set to zero.

uid: The **UID** specifying the meeting associated with the response.

sequence: The sequence number of a meeting instance associated with the response. The default value is zero.

utcDateTimeOrganizerCriticalChange: The time stamp at which the attendee responded to the meeting, translated to the time zone of the meeting organizer. [<1>](#)

utcDateTimeAttendeeCriticalChange: The time stamp at which the attendee responded to the meeting, translated to the time zone of the attendee.

response: Attendee acceptance or rejection of the meeting invitation. The default value is "responseAccepted".

3.1.4.9.2.2 SetAttendeeResponseResponse

The **SetAttendeeResponseResponse** element contains nothing other than standard SOAP faults if an error occurs. This element is defined as follows.

```
<s:element name="SetAttendeeResponseResponse">
  <s:complexType/>
</s:element>
```

3.1.4.9.3 Simple Types

3.1.4.9.3.1 AttendeeResponse

The **AttendeeResponse** element specifies the input parameters for the **SetAttendeeResponse** operation (section [3.1.4.9](#)). This element is defined as follows.

```
<s:simpleType name="AttendeeResponse">
  <s:restriction base="s:string">
    <s:enumeration value="responseAccepted"/>
    <s:enumeration value="responseTentative"/>
    <s:enumeration value="responseDeclined"/>
  </s:restriction>
</s:simpleType>
```

The following table specifies possible values for the **enumeration** attribute.

Value	Meaning
responseAccepted	The attendee has accepted the meeting invitation.
responseTentative	The attendee has tentatively accepted the meeting invitation.
responseDeclined	The attendee has declined the meeting invitation.

3.1.4.10 SetWorkspaceTitle

The **SetWorkspaceTitle** operation specifies a user-friendly name for a workspace. This operation is defined as follows.

```
<wsdl:operation name="SetWorkspaceTitle">  
  <wsdl:input message="SetWorkspaceTitleSoapIn" />  
  <wsdl:output message="SetWorkspaceTitleSoapOut" />  
</wsdl:operation>
```

The protocol client sends a **SetWorkspaceTitleSoapIn** request message (section [3.1.4.10.1.1](#)), and the protocol server responds with a **SetWorkspaceTitleSoapOut** response message (section [3.1.4.10.1.2](#)).

When a meeting workspace contains a single meeting instance and they have the same title, the meeting workspace title can be treated as the subject of the meeting instance. If the meeting organizer changes the title of a meeting instance and that instance is connected with a meeting workspace, the protocol client uses this operation to update the workspace title.

3.1.4.10.1 Messages

3.1.4.10.1.1 SetWorkspaceTitleSoapIn

The **SetWorkspaceTitleSoapIn** message requests the **SetWorkspaceTitle** operation (section [3.1.4.10](#)).

The SOAP action value of the message is defined as follows.

```
http://schemas.microsoft.com/sharepoint/soap/meetings/SetWorkspaceTitle
```

The SOAP body contains a **SetWorkspaceTitle** element (section [3.1.4.10.2.1](#)).

3.1.4.10.1.2 SetWorkspaceTitleSoapOut

The **SetWorkspaceTitleSoapOut** message is the response to the **SetWorkspaceTitle** operation (section [3.1.4.10](#)).

The SOAP body contains a **SetWorkspaceTitleResponse** element (section [3.1.4.10.2.2](#)).

3.1.4.10.2 Elements

3.1.4.10.2.1 SetWorkspaceTitle

The **SetWorkspaceTitle** element specifies the input parameters associated with the **SetWorkspaceTitle** operation (section [3.1.4.10](#)). This element is defined as follows.

```
<s:element name="SetWorkspaceTitle">
  <s:complexType>
    <s:sequence>
      <s:element name="title" type="s:string" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

title: The new title of the meeting workspace. This string has a maximum length of 255 characters. If this parameter is absent, the new title is the empty string.

3.1.4.10.2.2 SetWorkspaceTitleResponse

The **SetWorkspaceTitleResponse** element specifies the output of **SetWorkspaceTitle** operation (section [3.1.4.10](#)). This element is defined as follows.

```
<s:element name="SetWorkspaceTitleResponse">
  <s:complexType/>
</s:element>
```

If this operation is sent to a Web site that is not a meeting workspace, the response MUST be a SOAP fault with SOAP fault code "0x00000006".

3.1.4.11 UpdateMeeting

The **UpdateMeeting** operation updates the content of a meeting in a workspace. This operation is defined as follows.

```
<wsdl:operation name="UpdateMeeting">
  <wsdl:input message="UpdateMeetingSoapIn" />
  <wsdl:output message="UpdateMeetingSoapOut" />
</wsdl:operation>
```

The protocol client sends an **UpdateMeetingSoapIn** request message (section [3.1.4.11.1.1](#)), and the protocol server responds with an **UpdateMeetingSoapOut** response message (section [3.1.4.11.1.2](#)).

This operation MUST be used when updating meetings that are not based on the Gregorian calendar, because the iCalendar format as used in the **UpdateMeetingFromICal** operation (section [3.1.4.12](#)) does not support non-Gregorian recurring meetings. The implementer SHOULD use **UpdateMeetingFromICal** in all other instances.

3.1.4.11.1 Messages

3.1.4.11.1.1 UpdateMeetingSoapIn

The **UpdateMeetingSoapIn** message requests the **UpdateMeeting** operation (section [3.1.4.11](#)).

The SOAP action value of the message is defined as follows.

```
http://schemas.microsoft.com/sharepoint/soap/meetings/UpdateMeeting
```

The SOAP body contains an **UpdateMeeting** element (section [3.1.4.11.2.1](#)).

3.1.4.11.1.2 UpdateMeetingSoapOut

The **UpdateMeetingSoapOut** message contains the response to the **UpdateMeeting** operation (section [3.1.4.11](#)).

The SOAP body contains an **UpdateMeetingResponse** element (section [3.1.4.11.2.2](#)).

3.1.4.11.2 Elements

3.1.4.11.2.1 UpdateMeeting

The **UpdateMeeting** element specifies the input parameters for the **UpdateMeeting** operation (section [3.1.4.11](#)). This element is defined as follows.

```
<s:element name="UpdateMeeting">
  <s:complexType>
    <s:sequence>
      <s:element name="uid" type="s:string"/>
      <s:element name="sequence" type="s:unsignedInt" minOccurs="0"/>
      <s:element name="utcDateStamp" type="s:dateTime" minOccurs="0"/>
      <s:element name="title" type="s:string" minOccurs="0"/>
      <s:element name="location" type="s:string" minOccurs="0"/>
      <s:element name="utcDateStart" type="s:dateTime"/>
      <s:element name="utcDateEnd" type="s:dateTime"/>
      <s:element name="nonGregorian" type="s:boolean" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

uid: Specifies the meeting to update.

sequence: The updated sequence number of this meeting. The default value is zero.

utcDateStamp: The time at which the meeting is updated.

title: The updated subject of this meeting. This string has a maximum length of 255 characters.

location: The updated physical or virtual location in which this meeting is to take place. This string has a maximum length of 255 characters.

utcDateStart: The updated start time of this meeting.

utcDateEnd: The updated end time of this meeting. This time **MUST** be greater than or equal to **utcDateStart**.

nonGregorian: Specifies whether the updated meeting is not in the Gregorian calendar. The default value is FALSE.

3.1.4.11.2.2 UpdateMeetingResponse

The **UpdateMeetingResponse** element specifies the output of **UpdateMeeting** operation (section [3.1.4.11](#)). This element is defined as follows.

```
<s:element name="UpdateMeetingResponse">
  <s:complexType/>
</s:element>
```

If this operation is sent to a Web site that is not a meeting workspace, the response **MUST** be a SOAP fault with SOAP fault code "0x00000006".

3.1.4.12 UpdateMeetingFromICal

The **UpdateMeetingFromICal** operation updates content of a meeting in a workspace, based on a iCalendar object. This operation is defined as follows.

```
<wsdl:operation name="UpdateMeetingFromICal">
  <wsdl:input message="UpdateMeetingFromICalSoapIn" />
  <wsdl:output message="UpdateMeetingFromICalSoapOut" />
</wsdl:operation>
```

The protocol client sends an **UpdateMeetingFromICalSoapIn** request message (section [3.1.4.12.1.1](#)), and the protocol server responds with an **UpdateMeetingFromICalSoapOut** response message (section [3.1.4.12.1.2](#)).

This operation is used to update meetings. The **UpdateMeeting** operation (section [3.1.4.11](#)) MUST be used when adding meetings that are not based on the Gregorian calendar because the iCalendar format as used by this operation does not support non-Gregorian recurring meetings.

3.1.4.12.1 Messages

3.1.4.12.1.1 UpdateMeetingFromICalSoapIn

The **UpdateMeetingFromICalSoapIn** message requests the **UpdateMeetingFromICal** operation (section [3.1.4.12](#)).

The SOAP action value of the message is defined as follows.

```
http://schemas.microsoft.com/sharepoint/soap/meetings/UpdateMeetingFromICal
```

The SOAP body contains an **UpdateMeetingFromICal** element (section [3.1.4.12.2.1](#)).

3.1.4.12.1.2 UpdateMeetingFromICalSoapOut

The **UpdateMeetingFromICalSoapOut** message contains the response to the **UpdateMeetingFromICal** operation (section [3.1.4.12](#)).

The SOAP body contains an **UpdateMeetingFromICalResponse** element (section [3.1.4.12.2.2](#)).

3.1.4.12.2 Elements

3.1.4.12.2.1 UpdateMeetingFromICal

The **UpdateMeetingFromICal** element specifies the input parameters associated with the **UpdateMeetingFromICal** operation (section [3.1.4.12](#)). This element is defined as follows.

```
<s:element name="UpdateMeetingFromICal">
  <s:complexType>
    <s:sequence>
      <s:element name="icalText" type="s:string" minOccurs="0"/>
      <s:element name="ignoreAttendees" type="s:boolean" maxOccurs="1" minOccurs="1"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

icalText: Updated information about the meeting instance. This element MUST be present. This MUST be a valid iCalendar object as specified in [\[RFC2445\]](#). If this parameter is an empty string, the

response MUST be a SOAP fault with SOAP fault code "0x00000005". The total number of ATTENDEE elements in this parameter MUST be less than or equal to 254. If this parameter contains more than 254 ATTENDEE tags, the response MUST be a SOAP fault with SOAP fault code "0x0000000d".

ignoreAttendees: Specifies whether this is a scheduling-only update or whether it affects attendees. The protocol client SHOULD [<2>](#) NOT set this value to FALSE unless the protocol client address book is missing or has only partial details.

3.1.4.12.2 UpdateMeetingFromICalResponse

The **UpdateMeetingFromICalResponse** element specifies the output of the **UpdateMeetingFromICal** operation (section [3.1.4.12](#)). This element is defined as follows.

```
<s:element name="UpdateMeetingFromICalResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="UpdateMeetingFromICalResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="UpdateMeetingFromICal">
              <s:complexType>
                <s:sequence>
                  <s:element name="AttendeeUpdateStatus" type="tns:AttendeeUpdateStatus"/>
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

UpdateMeetingFromICalResult: The response XML consists of one element that contains information used to update a meeting instance. If the operation is not successful, this represents a SOAP fault. If the **icalText** value is empty or is not valid, the protocol server returns a SOAP fault with SOAP fault code "0x00000005".

If this operation is sent to a Web site that is not a meeting workspace, the response MUST be a SOAP fault with SOAP fault code "0x00000006".

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

4 Protocol Examples

When a meeting organizer on the protocol client creates a new meeting instance, the meeting organizer can add the new meeting instance to a new or existing meeting workspace.

If the meeting organizer indicates that a new meeting instance in a workspace is required and the meeting instance is required to be in an existing meeting workspace, the protocol client uses **GetMeetingWorkspaces** to obtain a list of existing meeting workspaces from the protocol server. To display information about an existing workspace, such as the language in which the data is to be displayed, the protocol client uses **GetMeetingsInformation**. The list of workspaces is displayed to the meeting organizer for selection. If the meeting organizer wants to create a new workspace, the protocol client uses **CreateWorkspace**.

If the meeting organizer cancels the workspace creation process and the workspace has already been created when the process is cancelled, the protocol client calls **DeleteWorkspace** to delete the newly created workspace. If the meeting organizer changes the subject in the meeting request before the meeting request is e-mailed to all the attendees for the first time, the protocol client uses **SetWorkspaceTitle** to rename the meeting workspace.

Once a workspace is available, whether an existing workspace or a newly created one, the protocol client uses **AddMeetingFromICal** to add the new meeting instance to the meeting workspace. If the meeting instance is a recurring meeting that does not use the Gregorian calendar, the protocol client uses **AddMeeting** instead. At this point, the meeting request is e-mailed to all attendees.

Each attendee receives the meeting request e-mail. When the attendee replies to the meeting request indicating acceptance or rejection of the invitation or indicating an update to a previous response, the protocol client uses **SetAttendeeResponse** to update the meeting workspace with attendance status.

If the meeting organizer later updates the meeting information, such as its time or location, the protocol client uses **UpdateMeetingFromICal** to update the meeting workspace information. If the meeting instance is a recurring meeting that does not use the Gregorian calendar, the protocol client uses **UpdateMeeting** instead.

When updating an existing meeting instance, the meeting organizer can disconnect that meeting from its meeting workspace, in which case the protocol client uses **RemoveMeeting** to delete that meeting instance from the workspace. The meeting organizer can perform the removal while the meeting request is open, which implies that the updated meeting request might be closed without being saved. If that happens, the protocol client uses **RestoreMeeting** to have the protocol server revert that meeting instance to its last persisted state.

4.1 Create a New Meeting Workspace and Manage Meetings

This scenario demonstrates how to create a new meeting workspace and manage a meeting within the new workspace.

4.1.1 Find Existing Meeting Workspaces

In this example, a meeting organizer wants to find existing meeting workspace subsites within a parent site. The meeting organizer specifically wants to find those workspaces that do not contain meetings and those that are single instance workspaces.

The protocol client sends the following request message.

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
```

```

    <GetMeetingWorkspaces xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <recurring>0</recurring>
    </GetMeetingWorkspaces>
  </soap:Body>
</soap:Envelope>

```

The protocol server returns the status value "Success" in the following response message that contains information regarding the existing meeting workspaces.

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetMeetingWorkspacesResponse
      xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <GetMeetingWorkspacesResult>
        <MeetingWorkspaces>
          <Workspace Url="http://www.example.com/Meeting Workspace 1" Title="Meeting
Workspace 1" />
          <Workspace Url="http://www.example.com/Meeting Workspace 2" Title="Meeting
Workspace 2" />
        </MeetingWorkspaces>
      </GetMeetingWorkspacesResult>
    </GetMeetingWorkspacesResponse>
  </soap:Body>
</soap:Envelope>

```

4.1.2 Query for Meeting Workspace Site Templates

The meeting organizer decides that they do not want to use an existing meeting workspace, but rather they want to create a new meeting workspace. The organizer wants to have a meeting workspace that manages the agenda and attendees for a single meeting.

The protocol client sends the following request message to find the appropriate site template to use to create the meeting workspace.

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetMeetingsInformation
      xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <requestFlags>4</requestFlags>
      <lcid>1033</lcid>
    </GetMeetingsInformation>
  </soap:Body>
</soap:Envelope>

```

The protocol server returns the status value "Success" in the following response message.

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetMeetingsInformationResponse
      xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <GetMeetingsInformationResult>
        <MeetingsInformation>

```

```

<ListTemplates>
  <Template
    Name="MPS#0"
    Title="Basic Meeting Workspace"
    Id="2"
    Description="A site to plan, organize, and capture the results of a
meeting. It provides lists for managing the agenda, meeting attendees, and documents."
    ImageUrl="http://www.example.com/_layouts/images/stmw.png" />
  <Template
    Name="MPS#1"
    Title="Blank Meeting Workspace" Id="2" Description="A blank meeting site to
customize based on your requirements."
    ImageUrl="http://www.example.com/_layouts/images/stbm.png" />
  <Template
    Name="MPS#2"
    Title="Decision Meeting Workspace"
    Id="2"
    Description="A site for meetings that track status or make decisions. It
provides lists for creating tasks, storing documents, and recording decisions."
    ImageUrl="http://www.example.com/_layouts/images/stdm.png" />
  <Template
    Name="MPS#3"
    Title="Social Meeting Workspace"
    Id="2"
    Description="A site to plan social occasions. It provides lists for
tracking attendees, providing directions, and storing pictures of the event."
    ImageUrl="http://www.example.com/_layouts/images/stsm.png" />
  <Template
    Name="MPS#4"
    Title="Multipage Meeting Workspace"
    Id="2"
    Description="A site to plan, organize, and capture the results of a
meeting. It provides lists for managing the agenda and meeting attendees in addition to
two blank pages to customize based on requirements."
    ImageUrl="http://www.example.com/_layouts/images/stmm.png" />
</ListTemplates>
</MeetingsInformation>
</GetMeetingsInformationResult>
</GetMeetingsInformationResponse>
</soap:Body>
</soap:Envelope>

```

4.1.3 Create a New Meeting Workspace

The meeting organizer determines that the MPS#0 site template is the appropriate one to use when creating the new meeting workspace.

The protocol client sends the following request message to create a new meeting workspace using the MPS#0 site template.

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <CreateWorkspace xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <title>Meeting Workspace 3</title>
      <templateName>MPS#0</templateName>
      <lcid>1033</lcid>
      <timeZoneInformation>
        <daylightDate>
          <year>2010</year>
          <month>3</month>
          <dayOfWeek>0</dayOfWeek>
          <day>14</day>

```

```

        <hour>2</hour>
        <minute>0</minute>
        <second>0</second>
        <milliseconds>0</milliseconds>
    </daylightDate>
</timeZoneInformation>
</CreateWorkspace>
</soap:Body>
</soap:Envelope>

```

The protocol server creates the new meeting workspace, and returns the status value "Success" in the following response message.

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <CreateWorkspaceResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <CreateWorkspaceResult>
        <CreateWorkspace Url="http://www.example.com/Meeting Workspace 3" />
      </CreateWorkspaceResult>
    </CreateWorkspaceResponse>
  </soap:Body>
</soap:Envelope>

```

4.1.4 Add a New Meeting

The meeting organizer wants to add a meeting to the workspace.

The protocol client sends the following request message to create the meeting in the new workspace.

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <AddMeeting xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <organizerEmail>someone@example.com</organizerEmail>
      <uid>A7E83F54-10DA-4dae-B548-4C9EA6129503</uid>
      <sequence>0</sequence>
      <utcDateStamp>2010-04-12T01:00:00Z</utcDateStamp>
      <title>The Title</title>
      <location>The Location</location>
      <utcDateStart>2010-04-15T01:00:00Z</utcDateStart>
      <utcDateEnd>2010-04-15T02:00:00Z</utcDateEnd>
      <nonGregorian>>false</nonGregorian>
    </AddMeeting>
  </soap:Body>
</soap:Envelope>

```

The protocol server creates the new meeting in the workspace, and returns the status value "Success" in the following response message.

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <AddMeetingResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <AddMeetingResult>

```

```

    <AddMeeting
      Url="http://www.example.com/Meeting Workspace 1?InstanceID=1"
      HostTitle="site1"
      UniquePermissions="true"
      MeetingCount="1"
      AnonymousAccess="false"
      AllowAuthenticatedUsers="false" />
  </AddMeetingResult>
</AddMeetingResponse>
</soap:Body>
</soap:Envelope>

```

4.1.5 Update the Meeting

The meeting organizer discovers that the time, location, and title of the meeting needs to be changed. To do this, the protocol client sends the following request message to update the meeting.

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <UpdateMeeting xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <uid>A7E83F54-10DA-4dae-B548-4C9EA6129503</uid>
      <sequence>0</sequence>
      <title>New Title</title>
      <location>New Location</location>
      <utcDateStamp>2010-04-12T08:00:00Z</utcDateStamp>
      <utcDateStart>2010-04-15T04:00:00Z</utcDateStart>
      <utcDateEnd>2010-04-15T05:00:00Z</utcDateEnd>
      <nonGregorian>false</nonGregorian>
    </UpdateMeeting>
  </soap:Body>
</soap:Envelope>

```

The protocol server updates the existing meeting, and returns the status value "Success" in the following response message.

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <UpdateMeetingResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/"
  />
  </soap:Body>
</soap:Envelope>

```

4.1.6 Respond to the Meeting

The meeting organizer wants to acknowledge that they have accepted the meeting and plan to attend it.

To accept the meeting, the protocol client sends the following request message.

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>

```

```

    <SetAttendeeResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <attendeeEmail>someone@example.com</attendeeEmail>
      <recurrenceId>0</recurrenceId>
      <uid>A7E83F54-10DA-4dae-B548-4C9EA6129503</uid>
      <sequence>0</sequence>
      <utcDateTimeOrganizerCriticalChange>2010-03-
15T01:05:00Z</utcDateTimeOrganizerCriticalChange>
      <utcDateTimeAttendeeCriticalChange>2010-03-
15T01:05:00Z</utcDateTimeAttendeeCriticalChange>
      <response>responseAccepted</response>
    </SetAttendeeResponse>
  </soap:Body>
</soap:Envelope>

```

The protocol server accepts the meeting on behalf of the organizer, and returns the status value "Success" in the following response message.

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <SetAttendeeResponseResponse
      xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/" />
  </soap:Body>
</soap:Envelope>

```

4.1.7 Remove the Meeting

The meeting organizer discovers that the meeting created is no longer needed and that it needs to be canceled.

To cancel the meeting, the protocol client sends the following request message.

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <RemoveMeeting xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <recurrenceId>0</recurrenceId>
      <uid>A7E83F54-10DA-4dae-B548-4C9EA6129503</uid>
      <sequence>0</sequence>
      <utcDateStamp>2010-04-12T10:00:00Z</utcDateStamp>
      <cancelMeeting>false</cancelMeeting>
    </RemoveMeeting>
  </soap:Body>
</soap:Envelope>

```

The protocol server cancels the existing meeting, and returns the status value "Success" in the following response message.

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <RemoveMeetingResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/"
  />
  </soap:Body>

```

```
</soap:Envelope>
```

4.1.8 Restore the Meeting

The meeting organizer decides that the meeting just canceled needs to be scheduled again.

To restore the meeting, the protocol client sends the following request message.

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <RestoreMeeting xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <uid>A7E83F54-10DA-4dae-B548-4C9EA6129503</uid>
    </RestoreMeeting>
  </soap:Body>
</soap:Envelope>
```

The protocol server restores the meeting, and returns the status value "Success" in the following response message.

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <RestoreMeetingResponse
      xmlns="http://schemas.microsoft.com/sharepoint/soap/meetings/" />
  </soap:Body>
</soap:Envelope>
```

5 Security

5.1 Security Considerations for Implementers

A protocol client can only receive information regarding workspaces to which the authenticated user has access, and which is permissible for the level of authorization on the meeting workspace or workspace parent. This issue affects the **GetMeetingWorkspaces** (section [3.1.4.6](#)) and **GetMeetingsInformation** (section [3.1.4.5](#)) operations.

5.2 Index of Security Parameters

The following table list the sections that describe the security parameters that are appropriate for this protocol.

Security parameter	Section
Transport authentication	Transport (section 2.1)

6 Appendix A: Full WSDL

For ease of implementation the full WSDL is provided in this appendix.

```
<?xml version="1.0" encoding="UTF-8" ?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tns="http://schemas.microsoft.com/sharepoint/soap/meetings/"
  xmlns:s="http://www.w3.org/2001/XMLSchema"
  xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
  xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
  targetNamespace="http://schemas.microsoft.com/sharepoint/soap/meetings/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <s:schema elementFormDefault="qualified"
      targetNamespace="http://schemas.microsoft.com/sharepoint/soap/meetings/">
      <s:import namespace="http://www.w3.org/2001/XMLSchema" />
      <s:simpleType name="CaseInsensitiveTrueFalse">
        <s:restriction base="s:string">
          <s:pattern value="[Tt][Rr][Uu][Ee]|[Ff][Aa][Ll][Ss][Ee]"/>
        </s:restriction>
      </s:simpleType>
      <s:simpleType name="CaseInsensitiveTrueFalseOrEmpty">
        <s:restriction base="s:string">
          <s:pattern value="([Tt][Rr][Uu][Ee]|[Ff][Aa][Ll][Ss][Ee])?"/>
        </s:restriction>
      </s:simpleType>
      <s:simpleType name="IntOrEmpty">
        <s:union memberTypes="s:int tns:Empty"/>
      </s:simpleType>
      <s:simpleType name="Empty">
        <s:restriction base="s:string">
          <s:maxLength value="0"/>
        </s:restriction>
      </s:simpleType>
      <s:element name="CreateWorkspace">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="title" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="templateName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="lcid" type="s:unsignedInt" />
            <s:element minOccurs="0" maxOccurs="1" name="timeZoneInformation"
              type="tns:TimeZoneInf" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:complexType name="TimeZoneInf">
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="bias" type="s:int" />
          <s:element minOccurs="1" maxOccurs="1" name="standardDate" type="tns:SysTime" />
          <s:element minOccurs="1" maxOccurs="1" name="standardBias" type="s:int" />
          <s:element minOccurs="1" maxOccurs="1" name="daylightDate" type="tns:SysTime" />
          <s:element minOccurs="1" maxOccurs="1" name="daylightBias" type="s:int" />
        </s:sequence>
      </s:complexType>
      <s:complexType name="SysTime">
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="year" type="s:unsignedShort" />
          <s:element minOccurs="1" maxOccurs="1" name="month" type="s:unsignedShort" />
          <s:element minOccurs="1" maxOccurs="1" name="dayOfWeek" type="s:unsignedShort" />
          <s:element minOccurs="1" maxOccurs="1" name="day" type="s:unsignedShort" />
          <s:element minOccurs="1" maxOccurs="1" name="hour" type="s:unsignedShort" />
          <s:element minOccurs="1" maxOccurs="1" name="minute" type="s:unsignedShort" />
          <s:element minOccurs="1" maxOccurs="1" name="second" type="s:unsignedShort" />
          <s:element minOccurs="1" maxOccurs="1" name="milliseconds" type="s:unsignedShort" />
        </s:sequence>
      </s:complexType>
    </s:schema>
  </wsdl:types>

```

```

<s:element name="CreateWorkspaceResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="CreateWorkspaceResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="CreateWorkspace"
              type="tns:CreateWorkspace" />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:complexType name="CreateWorkspace">
  <s:attribute name="Url" type="s:string" />
</s:complexType>
<s:element name="DeleteWorkspace">
  <s:complexType />
</s:element>
<s:element name="DeleteWorkspaceResponse">
  <s:complexType />
</s:element>
<s:element name="GetMeetingWorkspaces">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="recurring" type="s:boolean" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMeetingWorkspacesResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetMeetingWorkspacesResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="MeetingWorkspaces" minOccurs="0">
              <s:complexType>
                <s:sequence>
                  <s:element minOccurs="0" maxOccurs="unbounded" name="Workspace"
                    type="tns:Workspace" />
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:complexType name="Workspace">
  <s:attribute name="Url" type="s:string" />
  <s:attribute name="Title" type="s:string" />
</s:complexType>
<s:element name="SetWorkspaceTitle">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="title" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="SetWorkspaceTitleResponse">
  <s:complexType />
</s:element>
<s:element name="AddMeetingFromICal">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="organizerEmail" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="icalText" type="s:string" />
    </s:sequence>
  </s:complexType>

```

```

    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="AddMeetingFromICalResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="AddMeetingFromICalResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="AddMeetingFromICal" type="tns:AddMeetingFromICal" />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:complexType name="AddMeetingFromICal">
  <s:complexContent>
    <s:extension base="tns:AddMeeting">
      <s:sequence>
        <s:element name="AttendeeUpdateStatus" type="tns:AttendeeUpdateStatus" />
      </s:sequence>
    </s:extension>
  </s:complexContent>
</s:complexType>
<s:complexType name="AddMeeting">
  <s:attribute name="Url" type="s:string" />
  <s:attribute name="HostTitle" type="s:string" />
  <s:attribute name="UniquePermissions" type="s:boolean" />
  <s:attribute name="MeetingCount" type="s:int" />
  <s:attribute name="AnonymousAccess" type="s:boolean" />
  <s:attribute name="AllowAuthenticatedUsers" type="s:boolean" />
</s:complexType>
<s:complexType name="AttendeeUpdateStatus">
  <s:attribute name="Code" type="s:int" />
  <s:attribute name="Detail" type="s:string" />
  <s:attribute name="ManageUserPage" type="s:string" />
</s:complexType>
<s:element name="AddMeeting">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="organizerEmail" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="uid" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="sequence" type="s:unsignedInt" />
      <s:element minOccurs="0" maxOccurs="1" name="utcDateStamp" type="s:dateTime" />
      <s:element minOccurs="0" maxOccurs="1" name="title" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="location" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="utcDateStart" type="s:dateTime" />
      <s:element minOccurs="1" maxOccurs="1" name="utcDateEnd" type="s:dateTime" />
      <s:element minOccurs="0" maxOccurs="1" name="nonGregorian" type="s:boolean" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="AddMeetingResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="AddMeetingResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="AddMeeting" type="tns:AddMeeting" />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="UpdateMeetingFromICal">
  <s:complexType>
    <s:sequence>

```

```

        <s:element minOccurs="0" maxOccurs="1" name="icalText" type="s:string" />
        <s:element minOccurs="1" maxOccurs="1" name="ignoreAttendees" type="s:boolean" />
    </s:sequence>
</s:complexType>
</s:element>
<s:element name="UpdateMeetingFromICalResponse">
    <s:complexType>
        <s:sequence>
            <s:element name="UpdateMeetingFromICalResult" minOccurs="0">
                <s:complexType mixed="true">
                    <s:sequence>
                        <s:element name="UpdateMeetingFromICal">
                            <s:complexType>
                                <s:sequence>
                                    <s:element name="AttendeeUpdateStatus"
type="tns:AttendeeUpdateStatus"/>
                                </s:sequence>
                            </s:complexType>
                        </s:element>
                    </s:sequence>
                </s:complexType>
            </s:element>
        </s:sequence>
    </s:complexType>
</s:element name="UpdateMeeting">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="uid" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="sequence" type="s:unsignedInt" />
            <s:element minOccurs="0" maxOccurs="1" name="utcDateStamp" type="s:dateTime" />
            <s:element minOccurs="0" maxOccurs="1" name="title" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="location" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="utcDateStart" type="s:dateTime" />
            <s:element minOccurs="1" maxOccurs="1" name="utcDateEnd" type="s:dateTime" />
            <s:element minOccurs="0" maxOccurs="1" name="nonGregorian" type="s:boolean" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="UpdateMeetingResponse">
    <s:complexType />
</s:element>
<s:element name="RemoveMeeting">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="recurrenceId" type="s:unsignedInt"
/>
            <s:element minOccurs="1" maxOccurs="1" name="uid" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="sequence" type="s:unsignedInt" />
            <s:element minOccurs="0" maxOccurs="1" name="utcDateStamp" type="s:dateTime" />
            <s:element minOccurs="0" maxOccurs="1" name="cancelMeeting" type="s:boolean" />
        </s:sequence>
    </s:complexType>
</s:element name="RemoveMeetingResponse">
    <s:complexType />
</s:element>
<s:element name="SetAttendeeResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="attendeeEmail" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="recurrenceId" type="s:unsignedInt"
/>
            <s:element minOccurs="1" maxOccurs="1" name="uid" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="sequence" type="s:unsignedInt" />
            <s:element minOccurs="0" maxOccurs="1" name="utcDateTimeOrganizerCriticalChange"
type="s:dateTime" />
            <s:element minOccurs="0" maxOccurs="1" name="utcDateTimeAttendeeCriticalChange"
type="s:dateTime" />
        </s:sequence>
    </s:complexType>
</s:element>

```

```

        <s:element minOccurs="0" maxOccurs="1" name="response"
type="tns:AttendeeResponse" />
    </s:sequence>
</s:complexType>
</s:element>
<s:simpleType name="AttendeeResponse">
    <s:restriction base="s:string">
        <s:enumeration value="responseAccepted" />
        <s:enumeration value="responseTentative" />
        <s:enumeration value="responseDeclined" />
    </s:restriction>
</s:simpleType>
<s:element name="SetAttendeeResponseResponse">
    <s:complexType />
</s:element>
<s:element name="GetMeetingsInformation">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="requestFlags" type="s:unsignedInt"
/>
            <s:element minOccurs="0" maxOccurs="1" name="lcid" type="s:unsignedInt" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="GetMeetingsInformationResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetMeetingsInformationResult">
                <s:complexType mixed="true">
                    <s:sequence>
                        <s:element name="MeetingsInformation">
                            <s:complexType mixed="true">
                                <s:sequence>
                                    <s:element minOccurs="0" maxOccurs="1" name="AllowCreate"
type="tns:CaseInsensitiveTrueFalse" />
                                    <s:element name="ListTemplateLanguages" minOccurs="0">
                                        <s:complexType>
                                            <s:sequence>
                                                <s:element minOccurs="0" maxOccurs="unbounded" name="LCID"
type="s:string" />
                                            </s:sequence>
                                        </s:complexType>
                                    </s:element>
                                    <s:element name="ListTemplates" minOccurs="0">
                                        <s:complexType>
                                            <s:sequence>
                                                <s:element minOccurs="0" maxOccurs="unbounded"
name="Template" type="tns:Template" />
                                            </s:sequence>
                                        </s:complexType>
                                    </s:element>
                                    <s:element minOccurs="0" maxOccurs="1" name="WorkspaceStatus"
type="tns:WorkspaceStatus" />
                                </s:sequence>
                            </s:complexType>
                        </s:element>
                    </s:sequence>
                </s:complexType>
            </s:element>
        </s:sequence>
    </s:complexType>
</s:element>
<s:complexType name="Template">
    <s:attribute name="Name" type="s:string" />
    <s:attribute name="Title" type="s:string" />
    <s:attribute name="Id" type="s:int" />
    <s:attribute name="Description" type="s:string" />
    <s:attribute name="ImageUrl" type="s:string" />
</s:complexType>

```

```

    <s:complexType name="WorkspaceStatus">
      <s:attribute name="UniquePermissions" type="tns:CaseInsensitiveTrueFalseOrEmpty" />
      <s:attribute name="MeetingCount" type="tns:IntOrEmpty" />
      <s:attribute name="AnonymousAccess" type="tns:CaseInsensitiveTrueFalseOrEmpty" />
      <s:attribute name="AllowAuthenticatedUsers"
type="tns:CaseInsensitiveTrueFalseOrEmpty" />
    </s:complexType>
    <s:element name="RestoreMeeting">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="uid" type="s:string" />
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element name="RestoreMeetingResponse">
      <s:complexType />
    </s:element>
  </s:schema>
</wsdl:types>
<wsdl:message name="CreateWorkspaceSoapIn">
  <wsdl:part name="parameters" element="tns:CreateWorkspace" />
</wsdl:message>
<wsdl:message name="CreateWorkspaceSoapOut">
  <wsdl:part name="parameters" element="tns:CreateWorkspaceResponse" />
</wsdl:message>
<wsdl:message name="DeleteWorkspaceSoapIn">
  <wsdl:part name="parameters" element="tns>DeleteWorkspace" />
</wsdl:message>
<wsdl:message name="DeleteWorkspaceSoapOut">
  <wsdl:part name="parameters" element="tns>DeleteWorkspaceResponse" />
</wsdl:message>
<wsdl:message name="GetMeetingWorkspacesSoapIn">
  <wsdl:part name="parameters" element="tns:GetMeetingWorkspaces" />
</wsdl:message>
<wsdl:message name="GetMeetingWorkspacesSoapOut">
  <wsdl:part name="parameters" element="tns:GetMeetingWorkspacesResponse" />
</wsdl:message>
<wsdl:message name="SetWorkspaceTitleSoapIn">
  <wsdl:part name="parameters" element="tns:SetWorkspaceTitle" />
</wsdl:message>
<wsdl:message name="SetWorkspaceTitleSoapOut">
  <wsdl:part name="parameters" element="tns:SetWorkspaceTitleResponse" />
</wsdl:message>
<wsdl:message name="AddMeetingFromICalSoapIn">
  <wsdl:part name="parameters" element="tns:AddMeetingFromICal" />
</wsdl:message>
<wsdl:message name="AddMeetingFromICalSoapOut">
  <wsdl:part name="parameters" element="tns:AddMeetingFromICalResponse" />
</wsdl:message>
<wsdl:message name="AddMeetingSoapIn">
  <wsdl:part name="parameters" element="tns:AddMeeting" />
</wsdl:message>
<wsdl:message name="AddMeetingSoapOut">
  <wsdl:part name="parameters" element="tns:AddMeetingResponse" />
</wsdl:message>
<wsdl:message name="UpdateMeetingFromICalSoapIn">
  <wsdl:part name="parameters" element="tns:UpdateMeetingFromICal" />
</wsdl:message>
<wsdl:message name="UpdateMeetingFromICalSoapOut">
  <wsdl:part name="parameters" element="tns:UpdateMeetingFromICalResponse" />
</wsdl:message>
<wsdl:message name="UpdateMeetingSoapIn">
  <wsdl:part name="parameters" element="tns:UpdateMeeting" />
</wsdl:message>
<wsdl:message name="UpdateMeetingSoapOut">
  <wsdl:part name="parameters" element="tns:UpdateMeetingResponse" />
</wsdl:message>
<wsdl:message name="RemoveMeetingSoapIn">
  <wsdl:part name="parameters" element="tns:RemoveMeeting" />

```

```

</wsdl:message>
<wsdl:message name="RemoveMeetingSoapOut">
  <wsdl:part name="parameters" element="tns:RemoveMeetingResponse" />
</wsdl:message>
<wsdl:message name="SetAttendeeResponseSoapIn">
  <wsdl:part name="parameters" element="tns:SetAttendeeResponse" />
</wsdl:message>
<wsdl:message name="SetAttendeeResponseSoapOut">
  <wsdl:part name="parameters" element="tns:SetAttendeeResponseResponse" />
</wsdl:message>
<wsdl:message name="GetMeetingsInformationSoapIn">
  <wsdl:part name="parameters" element="tns:GetMeetingsInformation" />
</wsdl:message>
<wsdl:message name="GetMeetingsInformationSoapOut">
  <wsdl:part name="parameters" element="tns:GetMeetingsInformationResponse" />
</wsdl:message>
<wsdl:message name="RestoreMeetingSoapIn">
  <wsdl:part name="parameters" element="tns:RestoreMeeting" />
</wsdl:message>
<wsdl:message name="RestoreMeetingSoapOut">
  <wsdl:part name="parameters" element="tns:RestoreMeetingResponse" />
</wsdl:message>
<wsdl:portType name="MeetingsSoap">
  <wsdl:operation name="CreateWorkspace">
    <wsdl:input message="tns:CreateWorkspaceSoapIn" />
    <wsdl:output message="tns:CreateWorkspaceSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="DeleteWorkspace">
    <wsdl:input message="tns>DeleteWorkspaceSoapIn" />
    <wsdl:output message="tns>DeleteWorkspaceSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="GetMeetingWorkspaces">
    <wsdl:input message="tns:GetMeetingWorkspacesSoapIn" />
    <wsdl:output message="tns:GetMeetingWorkspacesSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="SetWorkspaceTitle">
    <wsdl:input message="tns:SetWorkspaceTitleSoapIn" />
    <wsdl:output message="tns:SetWorkspaceTitleSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="AddMeetingFromICal">
    <wsdl:input message="tns:AddMeetingFromICalSoapIn" />
    <wsdl:output message="tns:AddMeetingFromICalSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="AddMeeting">
    <wsdl:input message="tns:AddMeetingSoapIn" />
    <wsdl:output message="tns:AddMeetingSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="UpdateMeetingFromICal">
    <wsdl:input message="tns:UpdateMeetingFromICalSoapIn" />
    <wsdl:output message="tns:UpdateMeetingFromICalSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="UpdateMeeting">
    <wsdl:input message="tns:UpdateMeetingSoapIn" />
    <wsdl:output message="tns:UpdateMeetingSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="RemoveMeeting">
    <wsdl:input message="tns:RemoveMeetingSoapIn" />
    <wsdl:output message="tns:RemoveMeetingSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="SetAttendeeResponse">
    <wsdl:input message="tns:SetAttendeeResponseSoapIn" />
    <wsdl:output message="tns:SetAttendeeResponseSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="GetMeetingsInformation">
    <wsdl:input message="tns:GetMeetingsInformationSoapIn" />
    <wsdl:output message="tns:GetMeetingsInformationSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="RestoreMeeting">
    <wsdl:input message="tns:RestoreMeetingSoapIn" />

```

```

        <wsdl:output message="tns:RestoreMeetingSoapOut" />
    </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="MeetingsSoap" type="tns:MeetingsSoap">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http" />
    <wsdl:operation name="CreateWorkspace">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/CreateWorkspace"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="DeleteWorkspace">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/DeleteWorkspace"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetMeetingWorkspaces">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/GetMeetingWorkspaces"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SetWorkspaceTitle">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/SetWorkspaceTitle"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="AddMeetingFromICal">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/AddMeetingFromICal"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="AddMeeting">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/AddMeeting"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>

```



```

        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="UpdateMeetingFromICal">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/UpdateMeetingFromICal"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="UpdateMeeting">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/UpdateMeeting"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="RemoveMeeting">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/RemoveMeeting"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SetAttendeeResponse">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/SetAttendeeResponse"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetMeetingsInformation">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/GetMeetingsInformation"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="RestoreMeeting">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/RestoreMeeting"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
</wsdl:binding>

```

```

    <wsdl:binding name="MeetingsSoap12" type="tns:MeetingsSoap">
      <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" />
      <wsdl:operation name="CreateWorkspace">
        <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/CreateWorkspace"
style="document" />
          <wsdl:input>
            <soap12:body use="literal" />
          </wsdl:input>
          <wsdl:output>
            <soap12:body use="literal" />
          </wsdl:output>
        </wsdl:operation>
      <wsdl:operation name="DeleteWorkspace">
        <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/DeleteWorkspace"
style="document" />
          <wsdl:input>
            <soap12:body use="literal" />
          </wsdl:input>
          <wsdl:output>
            <soap12:body use="literal" />
          </wsdl:output>
        </wsdl:operation>
      <wsdl:operation name="GetMeetingWorkspaces">
        <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/GetMeetingWorkspaces"
style="document" />
          <wsdl:input>
            <soap12:body use="literal" />
          </wsdl:input>
          <wsdl:output>
            <soap12:body use="literal" />
          </wsdl:output>
        </wsdl:operation>
      <wsdl:operation name="SetWorkspaceTitle">
        <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/SetWorkspaceTitle"
style="document" />
          <wsdl:input>
            <soap12:body use="literal" />
          </wsdl:input>
          <wsdl:output>
            <soap12:body use="literal" />
          </wsdl:output>
        </wsdl:operation>
      <wsdl:operation name="AddMeetingFromICal">
        <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/AddMeetingFromICal"
style="document" />
          <wsdl:input>
            <soap12:body use="literal" />
          </wsdl:input>
          <wsdl:output>
            <soap12:body use="literal" />
          </wsdl:output>
        </wsdl:operation>
      <wsdl:operation name="AddMeeting">
        <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/AddMeeting"
style="document" />
          <wsdl:input>
            <soap12:body use="literal" />
          </wsdl:input>
          <wsdl:output>
            <soap12:body use="literal" />
          </wsdl:output>
        </wsdl:operation>
      <wsdl:operation name="UpdateMeetingFromICal">

```

```

    <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/UpdateMeetingFromICal"
style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="UpdateMeeting">
    <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/UpdateMeeting"
style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="RemoveMeeting">
    <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/RemoveMeeting"
style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetAttendeeResponse">
    <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/SetAttendeeResponse"
style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMeetingsInformation">
    <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/GetMeetingsInformation"
style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="RestoreMeeting">
    <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/meetings/RestoreMeeting"
style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
</wsdl:binding>
</wsdl:definitions>

```

7 Appendix B: Product Behavior

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

- Microsoft Office 2003
- The 2007 Microsoft Office system
- Microsoft Office 2010 suites
- Microsoft SharePoint Foundation 2010
- Windows SharePoint Services 2.0
- Windows SharePoint Services 3.0
- Microsoft SharePoint Foundation 2013
- Microsoft Office 2013
- Microsoft Office 2016
- Microsoft SharePoint Server 2016

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. 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 that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

[<1> Section 3.1.4.9.2.1](#): The Windows SharePoint Services 3.0 implementation of the protocol server ignores the **utcDateTimeOrganizerCriticalChange** element and uses only **utcDateTimeAttendeeCriticalChange**.

[<2> Section 3.1.4.12.2.1](#): In the 2007 Microsoft Office system client implementation, this value is set to **true** when the address book is missing or has only partial details. ATTENDEE tags in iCalendar objects, as specified in [\[RFC2445\]](#), cannot be generated if the attendee(s) e-mail addresses cannot be determined.

8 Change Tracking

This section identifies changes that were made to this document since the last release. 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.
- The removal of a document from the documentation set.

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

The revision class **Editorial** means that the 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 changes were introduced. Minor editorial and formatting changes may have been made, but the technical content of the document is identical to the last released version.

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.
- 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 dochelp@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
Z Appendix B: Product Behavior	Updated list of supported products.	Y	Content updated due to protocol revision.

9 Index

A

Abstract data model
[server](#) 17
[Add a new meeting example](#) 44
[AddMeeting complex type](#) 14
[AddMeeting operation](#) 18
[AddMeetingFromICal operation](#) 20
[Applicability](#) 10
[AttendeeUpdateStatus complex type](#) 14
[Attribute groups](#) 16
[Attributes](#) 16

C

[Capability negotiation](#) 10
[CaseInsensitiveTrue simple type](#) 15
[CaseInsensitiveTrueFalseOrEmpty simple type](#) 15
[Change tracking](#) 61
[Complex types](#) 14
[AddMeeting](#) 14
[AttendeeUpdateStatus](#) 14
[Create a new meeting workspace and manage meetings example](#) 41
[Create a new meeting workspace example](#) 43
[CreateWorkspace operation](#) 22

D

Data model - abstract
[server](#) 17
[DeleteWorkspace operation](#) 25
[Detail element](#) 13

E

Elements
[Detail](#) 13
[IcalText](#) 13
[Lcid](#) 13
[RecurrenceId](#) 13
[Sequence](#) 13
[UID](#) 13
[Empty simple type](#) 15
Events
[local - server](#) 40
[timer - server](#) 40
Examples
[add a new meeting](#) 44
[create a new meeting workspace](#) 43
[create a new meeting workspace and manage meetings](#) 41
[find existing meeting workspaces](#) 41
[overview](#) 41
[query for meeting workspace site templates](#) 42
[remove the meeting](#) 46
[respond to the meeting](#) 45
[restore the meeting](#) 47
[update the meeting](#) 45

F

[Fields - vendor-extensible](#) 11
[Find existing meeting workspaces example](#) 41
[Full WSDL](#) 49

G

[GetMeetingsInformation operation](#) 26
GetMeetingWorkspaces operation ([section 3.1.4.6](#) 30, [section 3.1.4.7](#) 31)
[Glossary](#) 7
[Groups](#) 16

I

[IcalText element](#) 13
[Implementer - security considerations](#) 48
[Index of security parameters](#) 48
[Informative references](#) 9
Initialization
[server](#) 17
[IntOrEmpty simple type](#) 15
[Introduction](#) 7

L

[Lcid element](#) 13
Local events
[server](#) 40

M

Message processing
[server](#) 17
Messages
[AddMeeting complex type](#) 14
[AttendeeUpdateStatus complex type](#) 14
[attribute groups](#) 16
[attributes](#) 16
[CaseInsensitiveTrueFalse simple type](#) 15
[CaseInsensitiveTrueFalseOrEmpty simple type](#) 15
[complex types](#) 14
[Detail element](#) 13
[elements](#) 12
[Empty simple type](#) 15
[enumerated](#) 12
[groups](#) 16
[IcalText element](#) 13
[IntOrEmpty simple type](#) 15
[Lcid element](#) 13
[namespaces](#) 12
[RecurrenceId element](#) 13
[Sequence element](#) 13
[simple types](#) 15
[syntax](#) 12
[transport](#) 12
[UID element](#) 13

N

[Namespaces](#) 12

[Normative references](#) 9

O

Operations

[AddMeeting](#) 18
[AddMeetingFromICal](#) 20
[CreateWorkspace](#) 22
[DeleteWorkspace](#) 25
[GetMeetingsInformation](#) 26
[GetMeetingWorkspaces](#) 30
[RemoveMeeting](#) 31
[RestoreMeeting](#) 33
[SetAttendeeResponse](#) 34
[SetWorkspaceTitle](#) 36
[UpdateMeeting](#) 37
[UpdateMeetingFromICal](#) 39
[Overview \(synopsis\)](#) 9

P

[Parameters - security index](#) 48
[Preconditions](#) 10
[Prerequisites](#) 10
[Product behavior](#) 60
Protocol Details
[overview](#) 17

Q

[Query for meeting workspace site templates example](#)
42

R

[RecurrenceId element](#) 13
[References](#) 8
[informative](#) 9
[normative](#) 9
[Relationship to other protocols](#) 10
[Remove the meeting example](#) 46
[Respond to the meeting example](#) 45
[Restore the meeting example](#) 47
[RestoreMeeting operation](#) 33

S

Security

[implementer considerations](#) 48
[parameter index](#) 48
[Sequence element](#) 13

Sequencing rules

[server](#) 17

Server

[abstract data model](#) 17
[AddMeeting operation](#) 18
[AddMeetingFromICal operation](#) 20
[CreateWorkspace operation](#) 22
[DeleteWorkspace operation](#) 25
[GetMeetingsInformation operation](#) 26
[GetMeetingWorkspaces operation](#) 30
[initialization](#) 17
[local events](#) 40
[message processing](#) 17

[RemoveMeeting operation](#) 31
[RestoreMeeting operation](#) 33
[sequencing rules](#) 17
[SetAttendeeResponse operation](#) 34
[SetWorkspaceTitle operation](#) 36
[timer events](#) 40
[timers](#) 17
[UpdateMeeting operation](#) 37
[UpdateMeetingFromICal operation](#) 39
[SetAttendeeResponse operation](#) 34
[SetWorkspaceTitle operation](#) 36
[Simple types](#) 15
[CaseInsensitiveTrueFalse](#) 15
[CaseInsensitiveTrueFalseOrEmpty](#) 15
[Empty](#) 15
[IntOrEmpty](#) 15
[Standards assignments](#) 11
Syntax
[messages - overview](#) 12

T

Timer events

[server](#) 40

Timers

[server](#) 17

[Tracking changes](#) 61

[Transport](#) 12

Types

[complex](#) 14

[simple](#) 15

U

[UID element](#) 13

[Update the meeting example](#) 45

[UpdateMeeting operation](#) 37

[UpdateMeetingFromICal operation](#) 39

V

[Vendor-extensible fields](#) 11

[Versioning](#) 10

W

[WSDL](#) 49