Cisco webex

Cisco WebEx™ TSP Bridge API Guide v3.4

Version 3.4

cisco

Copyright

Modified: October 9, 2008 10:02 am

WebEx Communications, Inc.
3979 Freedom Circle
Santa Clara, CA 95054

Preface

The WebEx Telephony API Guide covers the following key points related to specific information for TSPs (teleconferencing service providers) to integrate their audio conferencing network with WebEx's standard Telephony API.

- Key features of the Telephony API.
- Platform availability.
- TSP Telephony System Architectural overview.
- Implementation details including: performance and network / system requirements, and XML API messaging.
- Integration instructions with API reference.
- Status and Error Codes.
- Data definition agreements.

Audience

This guide provides TSP developers with the Telephony API definitions necessary to customize adapters that integrate audio bridges in your teleconferencing networks.

It also provides WebEx System Administrators the configuration information needed to support TSP teleconferencing bridges.

What's New in this Release

- Teleconference sessions can be extended beyond the meeting duration time use the Conference Keep-Alive feature.
- Additional global number support has been added.
- New functionality to handle connections between the WebEx TSP server and on-premise audio solutions has been added.

Section Content

The *WebEx Telephony API Guide* includes the following chapters:

- A history of the API and the related documentation.
- Chapter 1,"Telephony TSP Release" on page 1, explains Telephony TSP SDK key features and platform availability.

- Chapter 2, “Architecture” on page 5, discusses TSP architecture.
- Chapter 3, “Implementation Details” on page 9, provides the API implementation details and security features.
- Chapter 4, “Telephony Configuration” on page 29, shows changes resulting from the telephony configuration setup process on the applicable end user screens.
- Chapter 5, “Integration Instructions with API Reference” on page 37, provides the actual API reference, integration instructions, and a code example.
- Appendix A, “Code Definitions” on page 83, provides the status codes and error codes.
- Appendix B, “Data Definition Agreements” on page 85, provides the maximum number of messages in a single transaction.
- “Glossary” on page 117, contains a list of terms and their definitions.
- “Index” on page 123, contains cross-references of important sections.

Typographic Conventions

This guide uses the following typographical conventions:

- Key terms appear in **boldface** where they are first defined.
- Book titles, emphasized words, and variables appear in *italics*.
- Code examples, code elements, URLs, and file names appear in `Courier New`.
- Optional arguments in command syntax appear in brackets [...].
- Links, or cross-references, appear in WebExBlue.

Contacting WebEx

We value your comments regarding our software products and documentation. If you have suggestions or need further assistance, please contact us at:

WebEx Communications, Inc.
307 West Tasman Drive
San Jose, CA 95134

Toll-free phone: 1-877-50-WebEx (1-877-509-3239)
Tel: 1-408-435-7000
Fax: 1-408-435-7004
URL: `www.webex.com`
Email address: `info@webex.com`

API History

Feature	Release Number	Delivery Date
Updated the *W2A_CloseConference* for the Conference Keep-Alive feature. Added the Global Number Support feature.	3.4	9/5/2008
Added On-Premise Link functionality which includes the new *A2W_CreateConnection* and *W2A_RspCreateConnection* commands.	3.3.2	
Added Silent Join feature. The version 3.1 is backward compatible to earlier Business Suites.	3.1	
Added Active Speaker and Sub-conference features. The version 3.0.1 is backward compatible, and can function normally without changing any code in the adapter.	3.0.1	
Modified to detect a TSP server failure and restore communication using a secondary server.	2.5.3	7/10/2006
Modified *W2A_AuthenticateAccount*, *W2A_CreateAccount*, *W2A_CreateConference* to contain an *ExtUserID*.	2.3	2/26/2004
Added *W2A_DeleteAccount* and *A2W_RspDeleteAccount*. Extended the definition of Selection in *W2A_CreateConference*. Added additional elements *WbxHostID* and *WbxSiteID* in *W2A_CreateAccount*.	2.2	2Q 2003
Added the format definition for *TollFreeNum* and *TollNum*. Modified the incorrect spelling "*Extention*" to "*Extension*".	2.1	2Q 2003
Added *W2A_AuthenticateAccount*, *W2A_CreateAccount* and *W2A_UpdateAccount*. Extended *W2A_CreateConference*, *W2A_Callout*, *W2A_UpdateConference*, *A2W_NotifyUserEnter*, *A2W_NotifyUserChange*, and the corresponding error codes.	2.0	2Q 2003
Changed the definition of attribute Type for *A2W_RspGetBridgeInfo* to decimal. Added a description for *A2W_RspCreateConference* and *A2W_RspCallout*.	1.8	4Q 2002
Added optional attributes *ExtConfID* and *AttendeeID* in *W2A_RspNotifyUserEnter* upon request from TSP partners. (This is optional so that existing adapter programs will still work.). Added required action items for *W2A_Reset* for server recovery. Defined call type for *A2W_RspGetCallInfo*. Specified *CountryCode* and *Extension* as optional.	1.7	3Q 2002
Removed *ConfKey* attribute from *W2A_UpdateConference*.	1.6	3Q 2002
Added *AttendeeID* attribute for A*2W_NotifyUserEnter* for call-in, and user status code 2 for user leaving.	1.5	2Q 2002

Feature	Release Number	Delivery Date
Added error code "-13", *A2W_RespMessageError* and *W2A_RespMessageError*. Added *TransID* and *MsgIDSequence* definitions in Data Definition Agreements, and maximum number of messages inside a single transaction.	1.4	2Q 2002
Added error code "-12" ("Unrecognized new XML message received"), *W2A_UpdateConference* and *A2W_RspUpdateConference*. Moved *PhoneNum* in *W2A_Callout* from attribute to sub-element for easy parsing.	1.3	2Q 2002
Added feedback from customers. Fail-over design examples provided. Call sequence number was added for each call. Other miscellaneous changes added.	1.1	2Q 2002
Initial release	1.0	2Q 2002

Contents

List of Figures

List of Tables

1

Telephony TSP Release

This chapter provides an overview of the TSP (Teleconferencing Service Provider) release.The Meeting Center 5.0 version for TSP functionality offers a new API designed to enhance integration for your teleconferencing services.

1.1 Included in the Telephony SDK

The following are features that are included in the Telephony SDK:

- Meeting Center - TSP Web Service.
- Telephony API Documentation - English only.
- Sample Code and ReadMe's.
- Optional downloadable adapters (installation and setup).
- FAQ (Frequently Asked Questions).

1.2 Key API Features

Below is a list of key API features offered by the TSP.

- Telephony meeting creation.
- Optional email customizations.
- Start or join a teleconference when the WebEx meeting has started.
- Meeting window call control.
- TSP server failover.
- Identify active speaker.
- Create sub-conference.
- Allow Silent Join in a meeting.

- Conference Keep-Alive.
- Global Number Support.
- On-Premise Link

1.2.1 Meeting Creation

- Telephony meeting creation through current existing PHP APIs for Single Sign-On are compatible with the Telephony API.

You will be able to do the following:

- Configure sites for toll free call-in, toll call-in, domestic call-back, and international call-back.
- Schedule teleconferences based on the number of attendees per meeting.
- Schedule call-in or call-back per meeting.

1.2.2 Optional Email Customization

WebEx email templates and notification are available, but optional, allowing you to use your own email templates or email servers. You may request customization for up to two email templates.

1.2.3 Start or Join a Teleconference when the WebEx Meeting has Started

A teleconference session can be started from the **Meeting** window by selecting a function from the Participant menu.

1.2.4 Meeting Window Call Control

- Toll free call-in and toll call-in numbers are configurable in the **Site Administrator** screen, and are visually displayed in the **Meeting** window.
- Within an existing **Meeting** window the host or attendee can use the Meeting Manager to perform the following call control capabilities through the API to the adapter and onto the TSP's bridge network.
  - Associate caller with phone icon identifiable by name.
  - Mute and mute all function identifiable by a red / green icon.
  - Any attendee may select mute.
  - Expel attendees.
  - Phone out to another attendee during meeting.
  - Restrict access.

1.2.5 TSP Server Failover

In case of a TSP server failure, the WebEx meeting server obtains the conference key (meeting number) of the failed server and sends it to the secondary server. The secondary server uses this key to create a conference and inform the adapter. The adapter responds with *A2W_RspCreateConference* using the same conference key, and continues the meeting with the original functionality.

1.2.6 Identify Active Speaker

Identify Active Speaker allows meeting participants to see who is speaking. To identify the active speaker, the adapter notifies the WebEx TSP server that one or more callers has started or stopped talking. The TSP server then notifies the WebEx client to display the active talker(s) animation. There is no expected response (W2A messages) for these commands. The typical delay from the time a caller starts or stops speaking to when their status is reflected in the WebEx participant list is less than 2.5 seconds.

1.2.7 Create Sub-conference

The Training Center and Event Center have the ability to support a subset of participants in a separate session, essentially leaving the main group and entering a sub-conference. This functionally of WebEx Teleconferencing is also supported with a partner bridge.

1.2.8 Allow Silent Join in a Meeting

The WebEx Sales Center allows a Sales Manager to silently monitor sales sessions without disclosing his or her presence to the host or attendees. The WebEx client is notified to hide the Sales Manager's name, and the telephony bridge is notified to suppress the join-tone typically heard when joining a teleconference.

1.2.9 Conference Keep-Alive

The Conference Keep-Alive feature added in TSP 3.4 allows a teleconference session to be extended beyond the duration of the meeting. Existing audio integrations require an adapter code change to take advantage of this feature.

1.2.10 Global Number Support

This feature allows WebEx partners to display an unlimited number of global phone number to the end-user. These phone number are hosted by the partner on a separate web page.

1.2.11 On-Premise Link

The on-premise adapter is able to initiate a secure, persistent connection to the WebEx TSP server. This feature currently only supports integration with Cisco MeetingPlace.

1.3 Platform Functionality

The WebEx Telephony API is developed using open standard redundant server and network technologies. You may use the API independent of any OS or hardware constraints. Your adapter development is managed by your internal hardware, network and redundancy design choices.

2

Architecture

This chapter provides an overview of the TSP (teleconferencing service provider) architectural elements.

For more information about the TSP calls referenced in this chapter, refer to “Integration Instructions with API Reference” on page 37 of this guide.

The TSP includes a plug-in telephony adapter architecture allowing plug-in adapters to be developed for new telephony servers without customizing the underlying code. This allows you to integrate your teleconferencing bridge network.

TSP also includes standard bridge adapters you can order for your particular bridges. The initial set of available adapters includes a *Voyant Bridge version 2035* and a *Voyant 2313 Adapter*.

WebEx develops and maintains the standard adapters according to priorities. In cases where you have telephony server expertise and resources, you may elect to develop your own adapter to own and maintain yourself.

2.1 Key Architectural Features

Some of the key architectural features of the TSP Bridge are:

- XML-based.
- SSL (Secure Socket Layer) over IP cloud.
- Performance supports up to a 1344 port bridge (one chassis) with 350 simultaneous calls.
- Based on Meeting Center 6.0 technology.
- Works independently of WebEx URL API and XML API.
- Functions across the public Internet.

- Integrates with each user's network elements such as proxy servers, firewalls, security servers, VPN servers, and load balancers.
- Supports future deployments with dedicated leased lines between a TSP user's network and WebEx's WIN. If dedicated lease line connectivity is required, this can be considered a custom installation, configuration, or maintenance for time and materials.

2.2 Bandwidth Requirement

- T1 is desired and a 256K connection is considered minimum.

2.3 Basic Architecture

WebEx servers provide Web meeting services hosted by WebEx and located inside the WebEx network. Telephony bridges providing voice conferencing services are hosted by a TSP partner and are located inside the TSP partner's network.

WebEx servers talk to the bridge through API calls over the internet or a dedicated T1 connection with an HTTP/HTTPS protocol. Each side must have their own Web server installed to process the calls and requests. API messages passing between the two sides are XML formatted.

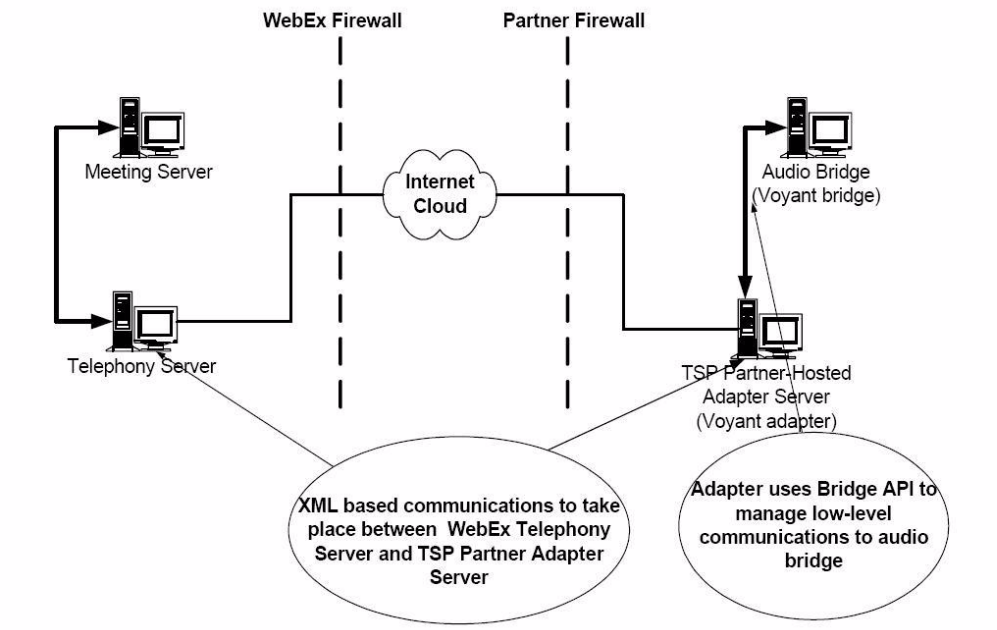
The API between WebEx and the TSP partner has a generic definition with basic telephony functionalities. Implementations of the functionalities are specific to the type of bridge used. Thus, an adapter application (or adapter) is required between the TSP partner Web server and the bridge to convert generic API calls to the bridge's own API calls.

The adapter's host name and HTTP port number are defined in the WebEx server configuration file by the WebEx System Administrator.

Messages initiated from the WebEx side are routed to the adapter using information found in the configuration file. Conference-related messages initiated from the adapter side are routed to the WebEx server using the URL given in a previously received *Create Conference* message. See “W2A_CreateConference” on page 47 for more information.

To achieve higher security, SSL (Secure Socket Layer) boxes can be placed in front of the Web server on each side.

Figure 2-1 • Architectural Model



Note The *Voyant* Adapter in the illustration above is an example of a WebEx application integrating with a *Voyant* Bridge.

2.4 System Architecture with Fail-over Feature

WebEx TSP architecture is designed to support different system fail-over and redundancy configurations. There are two parts to server redundancy:

- WebEx internal system redundancy
- TSP partner system redundancy

Two examples of redundancy implementation follow. The WebEx server supports both using different server configurations as specified in a configuration file.

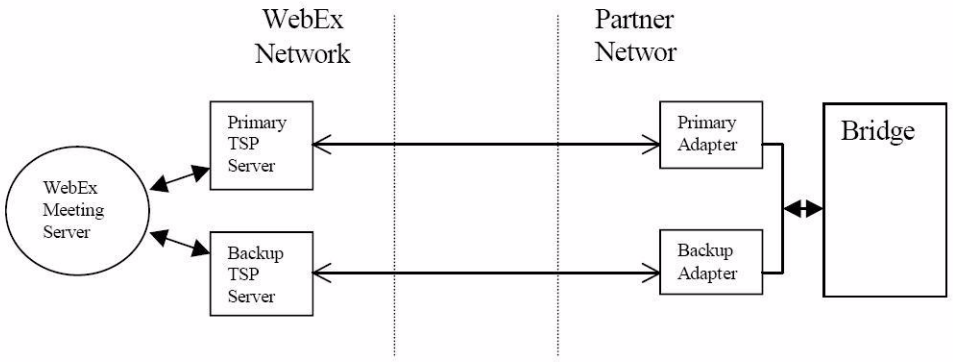
2.4.1 Transparent Redundancy Design

On the WebEx side, there are multiple application servers (i.e., TSP servers) running simultaneously, all of which communicate with the *single* adapter (or load balancer.) In the case where one TSP server goes down, traffic is automatically re-routed through the other servers.

In this design, the adapter's redundancy is transparent to WebEx. WebEx servers always communicate with the adapter(s) through a URL or host name. They do not need to know which adapter they are actually communicating with, and can consider them as a "single" adapter.

A load balancer can dynamically dispatch the WebEx requests to different adapters based on specified criteria. If one adapter fails, it will not be noticeable from the other end of the load balancer.

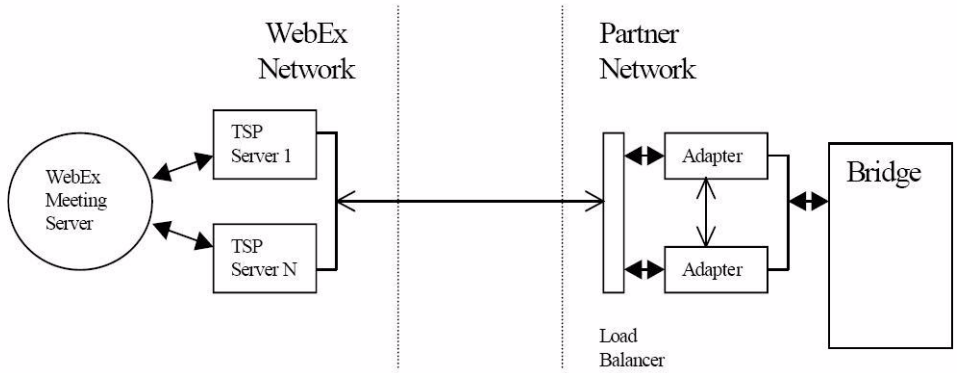
Figure 2-2 • Transparent Redundancy Design



2.4.2 Parallel Redundant Channels Design

This design is simpler. In a *Parallel Redundant Channels* design, each TSP server is configured to communicate with a dedicated adapter. The WebEx meeting server communicates with the bridge through the parallel channels. If a TSP server or an adapter goes down, the meeting server uses the other channel. No balancer is needed in this design.

Figure 2-3 • Parallel Redundant Channels



3

Implementation Details

This chapter provides an overview of the TSP (teleconferencing service provider) implementation details.

For more information about the TSP calls referenced in this chapter, refer to “Integration Instructions with API Reference” on page 37 within this guide.

3.1 Performance and Network / System Requirements

- XML API adapter performance supports at least 30 conferences per minute with a maximum of a 20-second delay per conference.
- After a telephony conference is initiated between the adapter and API, individual attendees do not experience more than a 10 second delay from the teleconference pop-up dialog box when joining a Web meeting.
- Bandwidth requirements and usage between the XML API and the adapter are specified as follows to enable the performance stated above:
  - A 256K connection is considered minimum bandwidth, T1 is recommended.

3.2 XML API Messaging Instructions

3.2.1 HTTP Request Target Machine

All XML messages are sent through HTTP requests. The following messaging considerations apply:

- For messages initiated from the WebEx side, the target machine should be the adapter whose URL for posting XML messages is given in the WebEx server configuration file and the WebEx System Administration tool.

- For adapter response messages to WebEx requests, the adapter should use the URL passed with the original request to find the target machine.
- Messages initiated from the adapter side (always associated with a conference) should be sent to the TSP server whose URL (for posting XML messages) is associated with the conference (i.e., passed from WebEx when sent a *W2A_CreateConference* request).

3.2.2 Logically Asynchronized Message Exchanges

Communication between the WebEx server and your adapter occurs through XML messaging defined as follows:

- One side sends a predefined XML message to the other side to request action or data using the HTTP post method.
- The receiving side checks the message to make sure the request is understood, and returns an OK confirmation.

However, the actual result of the action or data is returned in a separate message sent asynchronously to the requesting side.

Note If the original messages are predefined and known to the receiving side, the receiving side ensures all API parameters are given before an OK confirmation is sent. If an unknown message is found, an OK confirmation is still sent. But, an *Unrecognized new XML message received* error code -12 is returned separately as the result of that API message.

Thus, we are using asynchronized calls.

Note Each side should always use HTTP post to send a request message.

Here is an example of the call flow for creating a conference:

- WebEx sends the message *W2A_CreateConference* to an adapter using the HTTP post method, and waits for a response.
- The adapter receives the message and checks the format of the message against the predefined XML document definitions. If no error is found, it sends error code 0 (*OK*) in the response immediately. Otherwise, it sends a corresponding error code.
- Meanwhile, the adapter starts to call the *Bridge API* to create a conference, and waits for the conference creation confirmation from the bridge.

- After an interval of 2 seconds, the adapter receives a confirmation that the conference has been created in the bridge. It sends the message *A2W_RspCreateConference* to the WebEx server (the URL should have been part of the conference information stored in the adapter) by HTTP post, then waits for a response.
- WebEx receives the message, checks the message format, and sends the error code 0 (*OK*) back in response.

3.2.3 HTTP Transaction and API Messages

When using the HTTP post method, an HTTP transaction begins when the requesting side sends a request. It ends when a confirmation is received from the other end.

To obtain better performance in logical asynchronized calls, WebEx allows an HTTP request / transaction to send simultaneous, multiple XML API messages.

Note Therefore, the receiving side should check the formats of individual XML messages. Only when all of them meet requirements, is the error code 0 (OK) sent as a confirmation. Otherwise, a corresponding error code is returned.

The following is an example of an HTTP request (an XML document) that contains more then one XML API call:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <W2A_Callout
      MsgID="321" WbxHostName="tsp1.webex.com:80"
      ExtConfID="1234567" ExtCallID="123" Privilege="0"
      MaxWaitingTime="30">
      <PhoneNum>
         <CountryCode>1</CountryCode>
         <AreaCode>408</AreaCode>
         <LocalNumber>4357000</LocalNumber>
         <Extension>4508</Extension>
      </PhoneNum>
   </W2A_Callout>
   <W2A_CreateConference
      MsgID="322"
      WbxHostName="tsp1.webex.com:80" ExtConfID="1234568"
      ConfKey="987654321" MaxNumAttendee="50">
   </W2A_CreateConference>
</WbxTSPSchema>
```

When the adapter receives this document, it discovers two API calls:

- *W2A_Callout*
- *W2A_CreateConference*

The format of both elements is verified, and a Confirmation element is returned to WebEx if the format is acceptable. The two APIs then begin separately processing, and sending results to WebEx using other API calls.

3.2.4 Transaction ID, Resending, and API Message ID

Each HTTP transaction / request is given a *Transaction ID* (or a sequence number) that is always unique, except when the same request is issued more than once under time-out error conditions.

Similarly, each XML message is given a *Message ID*. The ID is used in each result message so the original requestor knows how to recognize the resulting request.

An important use of the *Transaction ID* is in message resending. Because WebEx communications use the Internet (where response times are not guaranteed), after sending an HTTP request, a WebEx sender may experience a time-out before a confirmation is received. In this case, the sender may retry an additional two times. Under these circumstances, the receiving side may receive the same call more than once.

Important Receivers should always check the *Transaction ID* of each request to detect duplicate requests. When duplicates are received, no further action should be taken except to return a confirmation. Normally, actions required by duplicate requests (i.e., duplicate IDs) should only be executed once by the receiver.

Transaction IDs are only reused when a previous request receives a time-out. Otherwise, a new ID should be used for each unique transaction request.

If a time-out occurs even after three attempts, rewrite the code to increase flexibility. Even if no TSP servers or adapters can be connected, the telephony conference and Web conference will continue to run assuming the conference was created. In other words, the other commands may time-out, but the conference will continue to run.

Figure 3-1 • TSP Request / Response

Typical TSP Request and Response

TSP Request Messages

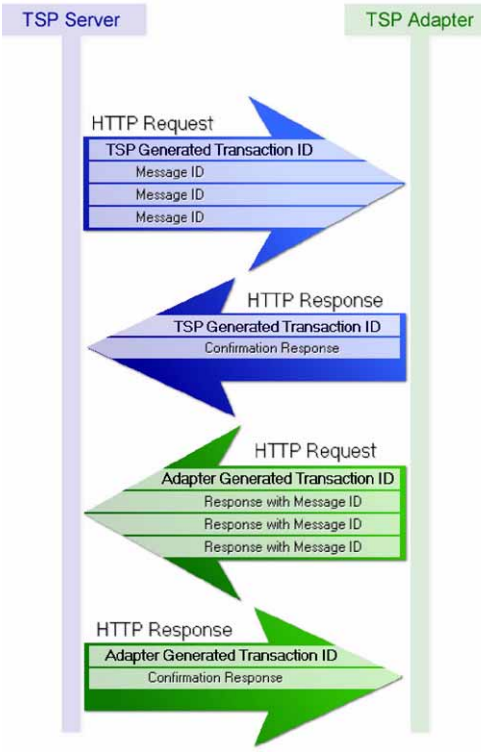
The TSP Server sends the Adapter a single transaction, containing three separate messages. The Transaction ID is generated by the TSP Server.

Confirmation Response

The adapter's HTTP response indicates that the transaction was received. This response contains the same Transaction ID as the request.

Asynchronous Response

The Adapter's response to each message is sent within a single transaction, containing a newly generated Transaction ID. Each Message ID, from the first transaction, is included here.



Confirmation Response

The TSP server's "confirmation response" is returned containing the same Transaction ID.

Note Adapters should not send *Close Conference* commands in response to HTTP disconnects.

3.2.5 Adapter Backward Compatibility

Existing APIs could be modified or extended in the future to support new features on the TSP server. To provide backward compatibility, adapters should ignore unknown attributes. When an adapter receives a new or unknown message, it should return "unknown msg received" using *A2W_RspMessageError*.

3.2.6 TSP Server Failover

In the event a TSP server fails, all meetings currently in progress are moved to another TSP server in the same pool.

Recognizing Failover

The WebEx meeting server regularly sends keep-alive messages to the TSP server to check its status. Should a meeting server receive errors or no response, it will attempt to create a TSP conference on another server.

The TSP Adapter will notice a failover scenario by the *ConfKey* (WebEx Meeting Number) that is passed in the *W2A_CreateConference* command sent by the secondary server; it will need to be able to recognize that the *ConfKey* passed is associated with a conference already in progress.

Failover Process

The following scenario describes how a typical failover will occur:

- The WebEx Meeting server notices that TSP Server 1 is not responding, it initiates a connection with TSP Server 2 and requests a new conference to be created, but with the same WebEx Meeting Number (*ConfKey*).
- TSP Server 2 sends the Partner Adapter a *W2A_CreateConference* command containing a new *ExtConfID* (teleconference unique identifier) and the same *ConfKey*.
- The adapter notices that the *ConfKey* sent is associated with a teleconference that is already in progress; using a different *ExtConfID*.
- The adapter replaces the existing *ExtConfID* with the new one provided. All subsequent commands relating to this conference / meeting will now use the new *ExtConfID*.

- The adapter replaces the existing *WbxHostName* with the new TSP server host name provided.
- The adapter responds with *A2W_RspCreateConference*, indicating that the conference was created successfully.
- The adapter sends its normal *A2W_NotifyConferenceChange* to begin the teleconference.
- The adapter sends all call-leg information to the TSP server by sending a transaction containing an *A2W_NotifyUserEnter* message for each participant.
- The TSP Server responds to each message by sending *W2A_RspNotifyUserEnter* messages.

Conference Features Affected During Failover

- If any of the call-legs were muted, the adapter will send *A2W_NotifyUserChange* messages to the TSP server in order to maintain state with the Meeting Manager.
- If the teleconference / meeting was locked prior to failover the adapter will need to make sure to maintain state when switching *ExtConfIDs*. The meeting manager will maintain its own lock-state separately.
- If additional participants joined prior to the TSP failover, then after the TSP server takes over the teleconference, the adapter must use *A2W_NotifyUserEnter* to notify the TSP server of all the new callers that joined.
- If participants left the teleconference during a TSP failover, then after the redundant TSP server takes over the teleconference, the adapter must use *W2A_NotifyUserChange* to notify the TSP server which participants left using the “status” code 3 (User Leave).
- If the teleconference closed during a TSP failover, the adapter must remember which conference closed. If the TSP server attempts to open a conference with the same *ConfKey*, the adapter must respond with error code “11” indicating a closed conference.
- If the host has initiated a private talk session, via the Invite by Phone feature, the adapter will need to ensure that it moves the invitee back into the main teleconference in a failover situation. This will be done by sending the *A2W_NotifyUserEnter* message for the invitee.
- In the event that a sub-conference is in progress, the Adapter will need to ensure that it notifies the WebEx TSP Server of the sub-conference. This is done by sending the *A2W_NotifySubConferenceChange* message with a *Status* code of “2”.

3.2.7 Identify Active Speaker

When a conference is created using *W2A_CreateConference*, the TSP server sends *W2A_UpdateConference* to the adapter using *ActionCode* 8 or 9:

- 8 is a notice to turn on the active speaker feature.
- 9 is a notice to turn off the active speaker notification feature.

The adapter must send the *A2W_NotifySpeakingStatus* message to the TSP server for the active speaker. The message must contain the *ConfID* and *CallerID* of the currently speaking participant. Because this command may send the speaking status of multiple meeting participants, a TSP server response is not expected for this command.

The following example demonstrates the Identify Active Speaker feature when both caller 1 and caller 2 start talking:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <A2W_NotifySpeakingStatus
      MsgID="321" ExtCallIDs="1%2" SpeakingFlags="Y%Y" ExtConfID=″xxx″>
   </A2W_NotifySpeakingStatus>
</WbxTSPSchema>
```

When caller 1 stops talking.

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <A2W_NotifySpeakingStatus>
      MsgID="321" ExtCallIDs="1" SpeakingFlags="N" ExtConfID=″xxx″>
   </A2W_NotifySpeakingStatus>
</WbxTSPSchema>
```

3.2.8 Sub-conference

The sub-conference feature is supported by WebEx Teleconferencing with a partner bridge. Essentially, this feature allows a subset of participants to leave the main group to enter a sub-conference.

TSP Server Support

The TSP server sends *W2A_CreateSubConference*, *W2A_CloseSubConference*, *W2A_AddToSubConference*, *W2A_RemoveFromSubConference* messages, depending on the type of action performed within the Training / Event Center.

The following is the XML syntax for creating a sub-conference:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <W2A_CreateSubConference>
      MsgID="322" WbxHostName="tsp1.webex.com:80"
      ExtConfID="1234568" ExtSubConfID="123“>
   </W2A_CreateSubConference>
</WbxTSPSchema>
```

The following is the XML syntax for adding callers to a sub-conference of the main conference:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
<TransID>121</TransID>
   <W2A_AddToSubConference
      MsgID="322" WbxHostName="tsp1.webex.com:80"
      ExtConfID="1234568" ExtSubConfID="123“ ExtCallID="123"
      ModeratorFlag="0">
   </W2A_AddToSubConference>
</WbxTSPSchema>
```

The following is the XML syntax for removing a set of callers from a sub-conference:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <W2A_RemoveFromSubConference
      MsgID="322" WbxHostName="tsp1.webex.com:80"
      ExtConfID="1234568" ExtSubConfID="123“
      ExtCallID="123">
   </W2A_RemoveFromSubConference>
</WbxTSPSchema>
```

The following is the XML syntax of closing a sub-conference:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <W2A_CloseSubConference
      MsgID="322" WbxHostName="tsp1.webex.com:80"
      ExtConfID="1234568" ExtSubConfID="123">
   </W2A_CloseSubConference>
</WbxTSPSchema>
```

Adapter Support

The adapter receives sub-conference requests from the TSP server and responds with *A2W_RspCreateSubConference*, *A2W_RspCloseSubConference*, *A2W_RspAddToSubConference*, and *A2W_RspRemoveFromSubConference* messages.

The following is XML syntax for the adapter's response to create a sub-conference:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <A2W_RspCreateSubConference
      MsgID="322" Result="0">
   </A2W_RspCreateSubConference>
</WbxTSPSchema>
```

The following is XML syntax for the adapter's response to adding callers to a sub-conference:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <A2W_RspAddToSubConference
      MsgID="322" Result="0">
   </A2W_RspAddToSubConference>
</WbxTSPSchema>
```

The following is XML syntax for the adapter's response of removing callers from a sub-conference:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <A2W_RspRemoveFromSubConference
      MsgID="322" Result="0">
   </A2W_RspRemoveFromSubConference>
</WbxTSPSchema>
```

The following is XML syntax for the adapter's response to closing a sub-conference:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <A2W_RspCloseSubConference
      MsgID="322" Result="0">
   </A2W_RspCloseSubConference>
</WbxTSPSchema>
```

Similar to a main conference, the adapter always notifies the TSP server whenever the adapter status changes, a caller joins the sub-conference, or there is a result from the adapter's action.

The following is XML syntax for notifying the TSP server of status changes for a sub-conference:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <A2W_NotifySubConferenceChange
      MsgID="322" ExtConfID="1234567"
      ExtSubConfID="123" Status="0">
   </A2W_NotifySubConferenceChange>
</WbxTSPSchema>
```

The following is XML syntax notifying the TSP server when a new caller joins the sub-conference:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <A2W_NotifySubConferenceChange
      MsgID="322" ExtConfID="1234567"
      ExtSubConfID="123" Status="2">
   </A2W_NotifySubConferenceChange>
   <A2W_NotifyJoinSubConference
      MsgID="322" ExtConfID="1234567"
      ExtSubConfID="123" ExtCallID="123" ModeratorFlag="0">
   </A2W_NotifyJoinSubConference>
</WbxTSPSchema>
```

Note This XML API is only used if a TSP server goes down when a conference fails-over to a new TSP server.

The following is XML syntax notifying the TSP server of a result to the adapter's action:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <A2W_NotifySubConferenceChange
      MsgID="322" ExtConfID="1234567"
      ExtSubConfID="123" Status="2"
   </A2W_NotifySubConferenceChange>
   <A2W_NotifySubConfCallChange
      MsgID="322" ExtConfID="1234567"
      ExtSubConfID="123" ActionFlag="1" Status="0" ExtCallID="123">
   </A2W_NotifySubConfCallChange>
</WbxTSPSchema>
```

Note This XML API is only used if a TSP server goes down when a conference fails-over to a new TSP server.

Note There is no need for specific error codes for sub-conference related errors. Error code, "-1", is a general error in the sub-conference XML API.

3.2.9 Silent Join

WebEx Sales Center allows Sales Managers to silently monitor sales sessions without the host or attendees being aware of their presence. With Silent Join, the Sales Manager's name is not displayed, and the audio portion suppresses the join-tone typically heard when joining a teleconference. Sales Managers can monitor both call-out and call-in teleconferencing using this feature.

Call-out Teleconferencing

When a user enters his phone number into a Sales Center Client, he will receive a call back. The TSP server's *W2A_Callout* message is used to do this. This message contains the *ParticipantType* attribute containing a value of "6" (silent monitoring attendee). To support Silent Join, the Adapter / Bridge must suppress the join-tone and mute the caller when the ParticipantType attribute of "6" is received.

The following is XML syntax using Silent Join in call-out teleconferencing:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>121</TransID>
   <W2A_Callout
      MsgID="321" WbxHostName="tsp1.webex.com:80"
      ExtConfID="1234567" ExtCallID="123" Privilege="1"
      MaxWaitingTime="30" ParticipantType="6" JoinMode="1">
      <PhoneNum>
         <CountryCode>1</CountryCode>
         <AreaCode>408</AreaCode>
         <LocalNumber>4357000</LocalNumber>
         <Extension>4508</Extension>
      </PhoneNum>
      <Result>0</Result>
   </W2A_Callout>
</WbxTSPSchema>
```

Note For Silent Join to work in a call-out teleconference, a Sales Representative and his prospects must be in a WebEx Sales Session. Moreover, the session must use TSP integrated audio and all of the session attendees must be dialed into it.

Call-in Teleconferencing

When a Sales Session is started, the TSP server will send a *W2A_CreateConference* command, much like it does with any session; however, this command will contain a new attribute called *AnonymousID*. The Adapter needs to store this ID for later use, in the event that a Silent Monitoring is used.

When the Sales Manager joins his Sales Session he receives dialing instructions. The last line of the instructions prompts him to enter an Attendee ID number. For silent monitoring, the Attendee ID he enters via the phone will need to match the *AnonymousID* that was passed in the *W2A_CreateConference* message. The Adapter will now be able to ensure the bridge's join tone is suppressed and to mute call-leg.

The following is XML syntax for Silent Join to a call-in teleconferencing:

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>122</TransID>
   <A2W_NotifyUserEnter MsgID="322" WbxHostName="tsp1.webex.com:80"
      ExtConfID="1234567" ExtCallID="null" AttendeeID="444"
      ActionResult="0" ParticipantType="6">
   </A2W_NotifyUserEnter >
</WbxTSPSchema>
```

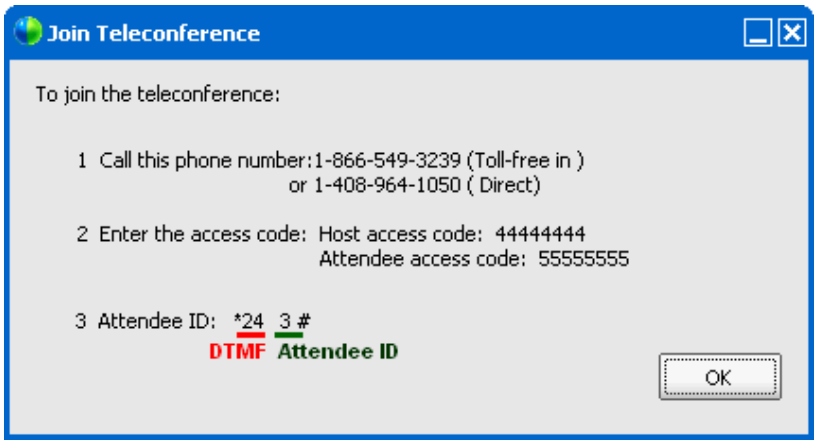
Note For Silent Monitoring to work with call-in teleconferencing, the bridge has to be able to prompt for an Attendee ID. Example: a voice saying, "Please enter your attendee ID number". Silent Join for call-in teleconferencing is not supported when using the Call-in User Merge feature, using DTMF codes to pass the attendee ID will occur after the sales manager is already in the conference.

Call-in User Merge

During the final stage of dialing into a WebEx teleconferencing bridge, users are prompted to enter an attendee ID. The ID is passed from the bridge to the WebEx client. This places a phone icon next to the user's name and links the call leg to the WebEx attendee. Most Telco Partners are not willing to prompt their users to enter an Attendee ID, as a third stage of dialing, which results in a duplicate entry in the participant panel, called "Call-in User X". Additionally, no phone icon appears next to the user's name in the participant panel. This renders the Call-in Teleconferencing option virtually useless to Telco Partners.

The Call-in User Merge feature solves this problem. Call-in User Merge allows Telco Partners to pass the attendee ID via DTMF code, rather than prompting the user via the audio. The WebEx Meeting Manager prompts the user to enter the DTMF code, followed by the attendee ID.

Figure 3-2 • Join Teleconference.



Instead of requiring the bridge to say, "Please enter your attendee ID", the WebEx client's "Join Teleconference" box prompts the user to perform step three.

Network Based Recording Support

NBR in WBS26 has the ability to dial out to the TSP integrated bridge and connect to the teleconference automatically. It does this by using the TSP account information selected when the meeting was scheduled. NBR dials into the teleconference bridge just as a participant would. This creates a scenario where a "Call-in User x" entry is added to the participant list and can cause confusion. Some teleconferencing providers are not concerned by this additional entry, but a method has been designed to allow its removal.

The Call-in User Merge feature, explained above, is instrumental in removing the "Call-in User x" entry from the participant panel. The NBR dialer enters the DTMF code, followed by the Attendee ID, just as a participant would. This results in the Adapter passing the Attendee ID to the WebEx TSP server. The TSP server passes the merge instruction to the meeting server, which essentially removes the entry from the participant list and places a telephone icon next to the participant's name.

Enhanced Mute-upon-entry Support

Typically, when a large event’s host schedules a meeting, the ability to mute participants at join-time is very important. WebEx versions below WBS26 require the participant's client to send an individual mute command to the bridge and rely on the client to maintain state. This logic created scenarios that allowed some noise to get through before the mute command was executed. Very large events increased the possibility of disturbances exponentially.

Enhanced Mute-upon-entry resolves this issue by allowing the TSP server to maintain the mute state of each call leg. When the host enables the Mute-upon-entry feature, a single command is sent to the TSP server, allowing it to automatically mute each participant at join time. The enhancement dramatically reduces undesirable noises at join-time.

3.2.10 Conference Keep-Alive

The Conference Keep-Alive feature introduced in TSP 3.4 allows a teleconference session to be extended beyond the initial duration of a meeting.

- Existing audio integrations require an adapter code change to take advantage of this feature.
- The feature is only supported on Meeting Center.

API Changes

When an audio integrated meeting ends, the WebEx TSP Server sends the Partner Adapter the *W2A_CloseConference* command. In TSP 3.4, this command adds the new attribute, `KeepAlive` to the API function. The `KeepAlive` attribute is a boolean function with the following value definitions:

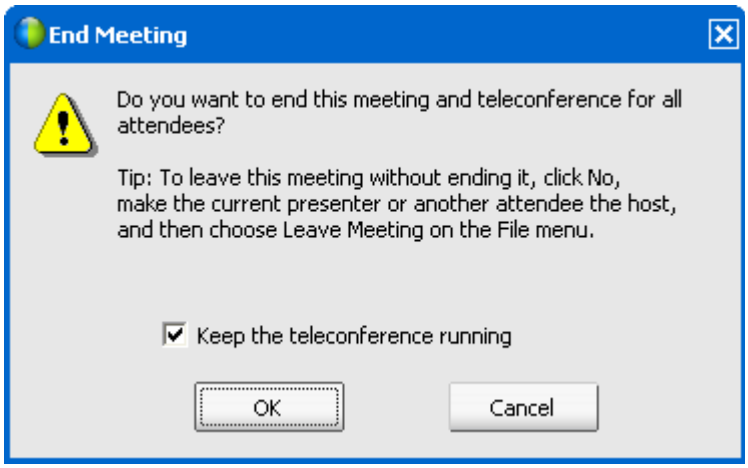
- 0 - Terminate the teleconference.
- 1 - Keep the teleconference alive.

The `KeepAlive` attribute is not required and will not be present in the XML if the Conference Keep-Alive feature is disabled, or not elected at meeting end-time. Users upgrading to or developing TSP 3.4 may opt-out of this feature, as the `KeepAlive` attribute is only included in the XML if `KeepAlive = 1`.

WebEx Configuration

When a meeting ends, a pop up window appears asking if the user wants to end the meeting for all attendees.

Figure 3-3 • End Meeting Request

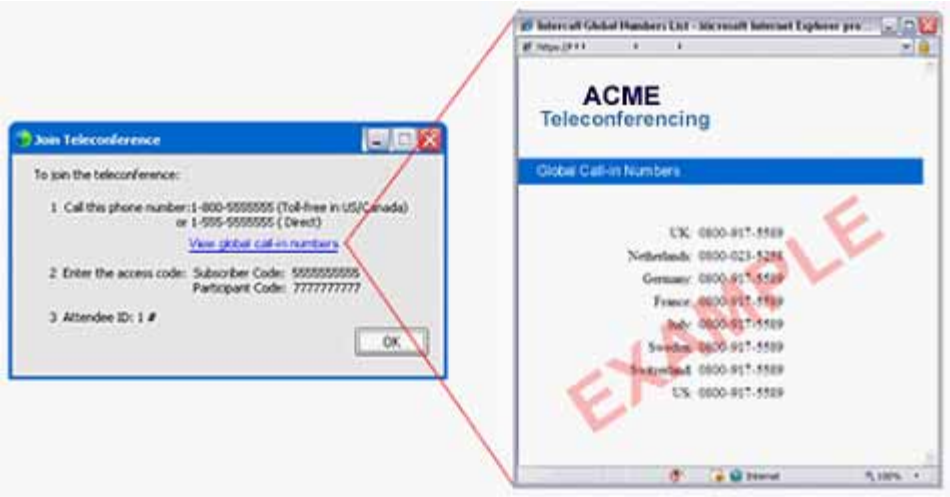


This feature can be enabled or disabled within the Site Admin interface. When the feature is disabled, the adaptor will not receive the `KeepAlive` attribute included in the *W2A_CloseConference* message. The default checkbox status of the pop up is configurable as a sub-item.

3.2.11 Global Number Support

WebEx partners are able to display an unlimited number of global phone numbers to end-users by hosting a web service/page that displays the numbers. WebEx embeds the link to those numbers within the client, email invites, and various other pre-meeting locations.

Figure 3-4 • Global Phone Number Example



The URL takes the end-user to the phone number page stored within the WebEx Telephony domain, controlled internally by WebEx. The provisioning of this URL must be coordinated with the API Developer Services team, or a WebEx Account Manager.

URL Variables

In order to provide customized content to the end-user, the telephony partner likely requires certain arguments to be passed to their web service from WebEx. The configured URL is dynamically created to pass any piece of the teleconference account information associated with the meeting. This is performed by utilizing variables within the URL string.

- Example of a URL at the time of configuration:
  - https://www.acme.com/globalnumbers/?subscribercode=%SubscriberCode%
- Example of the URL at scheduling time:
  - https://www.acme.com/globalnumbers/?subscribercode=12341234

The optional variables that can be used in a URL are shown in Table 3-1.

Table 3-1 • URL Variables

Name	Description
%ParticipantCode%	The participant code for the TSP account in use.
%SubscriberCode%	The subscriber code for the TSP account in use.
%TollFreeNum%	Toll-free number from the TSP account in use.
%TollFreeNumCC%	The country code for the toll-free number.
%TollNum%	Toll number from the TSP account in use.
%TollNumCC%	The country code for the toll number.

3.2.12 On-Premise Link

The on-premise adapter is able to initiate a secure, persistent connection to the WebEx TSP server, however WebEx cannot initiate a connection to an adapter behind a customer's firewall.

In order for a TSP server to work in OPL mode, it first needs to be configured to run in OPL mode. A server has the option of running in OPL Mode or Service Provider Mode, but not both. Adapters running in Service Provider Mode do not require any changes, as they do not utilize OPL functionality. Adapters built for on-premise solutions require the ability to initiate a secure connection to a WebEx TSP server as outlined in the following sections.

Adapter Configuration

The on-premise adapter needs to store the following WebEx related settings in order to make a successful OPL connection:

- WebEx Site Name - This is the sub-domain name of the WebEx site. For example: [sitename].webex.com.
- WebEx Partner ID - This acts as another level of security for integrating into the various WebEx APIs.

- Site Admin Username - A WebEx host account username with Site Administrator privileges that will be used for the XML and TSP APIs.

SAML Authentication

SAML is the sole means of authentication when making an OPL connection. The on-premise adapter must be able to generate a Base64 assertion which is used for two things:

- Authenticating against the XML API to retrieve a session ticket.
  - XML Command: AuthenticateUser
  - XML URL: https://[sitename].webex.com/WBXService/XMLService
  - The session ticket will replace the host account password for all XML API requests the adapter makes.
- TSP Server Connection (OPL)
  - A Base64 assertion will be sent via the *W2A_CreateConnection* message.

TSP Server IP Address Retrieval

The XML API `GetSite` command gives the on-premise adapter the ability to query WebEx for the TSP Server IP addresses to connect to. This command is best issued directly after the adapter has retrieved the XML API session ticket via the `AuthenticateUser` command.

Persistent HTTPS Connection

Using *A2W_CreateConnection*, the on-premise adapter needs to create a persistent connection that is used for bidirectional TSP traffic. The TSP *W2A_GetResource* message will be used as a "keep alive" message.

The adapter needs to make two connections, one to the primary server and the other to the secondary server, which will be used for failover.

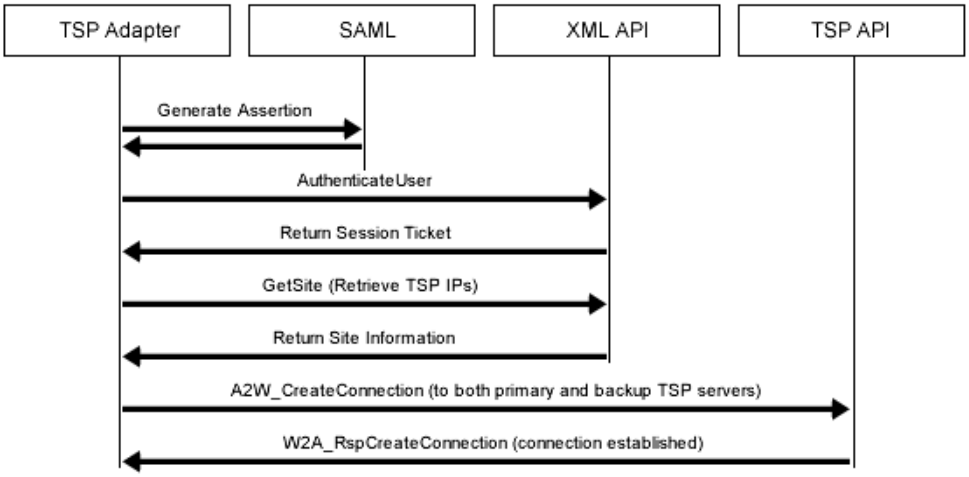
OPL Connection Flow

The following sequence shows the steps to create an on-premise adapter connection:

- TSP adapter generates the SML assertion to be used for TSP and XML API authentication.
- The TSP adapter authenticates against the WebEx XML API service.
- The XML API responds synchronously and provides a session ticket.
- The TSP adapter pulls WebEx site information, including TSP server IP addresses, by using the XML API `GetSite` command.

- The XML API responds synchronously to provide site information via the `GetSiteResponse` message.
- The TSP adapter uses the TSP IPs returned from `GetSiteResponse` to request a persistent connection to each server (primary and backup). The TSP command used to initiate the connection is *W2A_CreateConnection*.
- Each TSP server responds asynchronously with *A2W_RspCreateCreation*, with a a `Status` of 0 indicating a successful connection.

Figure 3-5 • On-Premise Link Diagram



4

Telephony Configuration

This chapter describes the changes on the applicable end user screens resulting from the telephony configuration setup process.

4.1 Telephony Configuration

Telephony configuration provides the input for the server configuration file. The WebEx System Administrator that provisions TSP partner sites completes this online form when the site is initially set up. It can be updated later as required. The configuration settings made during the initial setup become the new default settings.

TSP partners must provide the required information that establishes many settings. These are noted in the sections below.

Note Although TSP partners do not have direct access to edit or change these configurations, they may request updates.

4.2 Add a New Telephony Domain

The WebEx System Administrator must add a new telephony domain to set up your service. You must provide the following call-in numbers for domain setup:

- Primary large telephony server call-in numbers.
- Primary small telephony server call-in numbers.

Refer to the table below for field descriptions.

Table 4-1 • New Telephony Domain Field Description

Fields			Description
Primary Large Telephony Server	Toll call-in number		Call-in number with area code. Required field entry.
	Toll-free call-in number		Call-in number with area code. Optional field entry.
	Backup 1 Large Server	Toll call-in number	Call-in number with area code. Recommended backup bridge.
		Toll-free call-in number	Call-in number with area code. Recommended backup bridge.
	Backup 2 Large Server	Toll call-in number	Call-in number with area code.
		Toll-free call-in number	Call-in number with area code.

Primary Large Telephony Server

Large servers are for meetings requiring 15 or more attendees. The following field entries are required:

- Provide the *toll call-in number*.
- A *Toll-free call-in number* may be added, if applicable.
- Space is provided to add two backup servers. Using at least one large backup server is recommended.

Primary Small Telephony Server

Small servers are for meetings requiring less than 15 attendees. Refer to the descriptions in *Primary Large Telephony Server* field descriptions to provide these call-in numbers.

After all settings are made, your Telephony Domain is configured and the Telephony Domain Name appears on the appropriate WebEx lists.

4.2.1 Update a Site

The WebEx System Administrator will provision your site with the following Telephony configuration information you have provided.

- If you require a specific Teleconferencing service name, advise your WebEx administrator.

- Provide the primary and secondary telephony adapter URL information. Other telephony configuration information is detailed below.

Table 4-2 • Telephony Configuration Screen Field Description

Telephony Configuration ScreenFields	Description
Call-back teleconferencing	If selected, the teleconferencing service can call the user back to avoid a toll call.
Call-in teleconferencing	If selected, allows users to call in to a meeting on a phone line.
Default maximum number of callers	Default value for the estimated number of callers in the Schedule a Meeting on page 32. 16 character field limit.
International call-back teleconferencing	If selected, the teleconferencing service can call the user back to avoid an international toll call.
Internet Phone (VoIP)	Voice-over IP capabilities.
Major Area or city code	The major city code 5 character field limit.
Major country code	Select the telephony bridge country code from the drop-down menu.
Primary telephony adapter URL	URL of your adapter that communicates with the WebEx Telephony API, e.g., tsp.net/webexadapter. 64 character field limit.
Secondary telephony adapter URL	URL of your adapter that communicates with the WebEx Telephony API. 64 character field limit.
Teleconferencing service name	Brandable name for the teleconferencing service, e.g., My Teleconference. 64 character field limit.
Toll free call-in teleconferencing	If selected, allows users to call in using a toll-free 800 number.

The WebEx System Administrator updates your settings with the details you provide.

4.3 End User Screens

Once the WebEx System Administrator completes the telephony configuration and adds a new telephony domain, you will see the resulting end user screens:

- Schedule a Meeting.
- Teleconferencing pop-up screen.
- Participant menu.
- Participant icon display in the Meeting window.

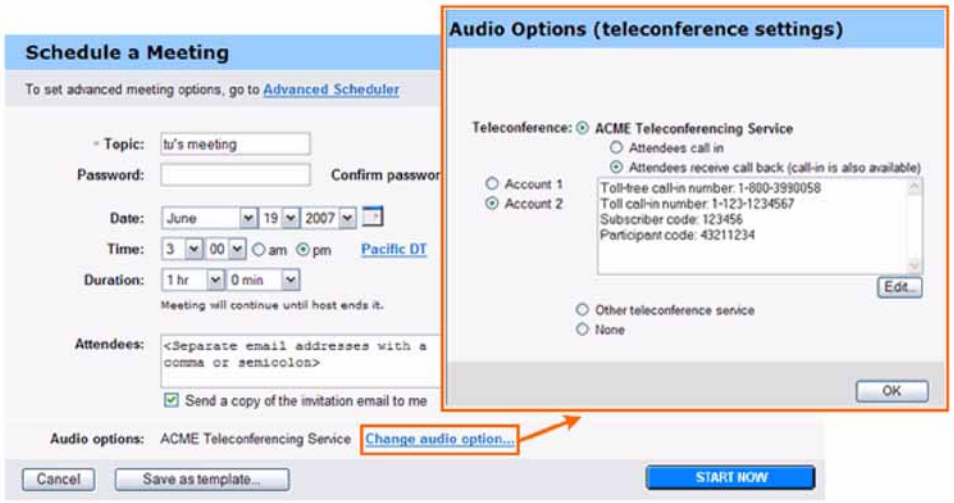
4.3.1 Schedule a Meeting

To schedule a meeting, use a Web browser to go to the site URL. Follow the steps below.

- From the navigation bar, go to the *Host a Meeting* section. Click Schedule a Meeting.
- At the Log In screen, enter the *User Name* and *Password*, and click Log In.
- The *Teleconference* section is the screen portion that changed after adding the telephony configuration and domain.

The Teleconference section of the Schedule a Meeting screen is displayed below:

Figure 4-1 • Teleconference Screen Section



Instead of the WebEx teleconferencing service, the screen now displays the teleconferencing service name the WebEx System Administrator configured.

- Select the teleconferencing service radio button, e.g.,*Teleconferencing Service Providers* displayed in the example above.
- Another branded example might be: *My Teleconference*.
- Click Start Meeting.

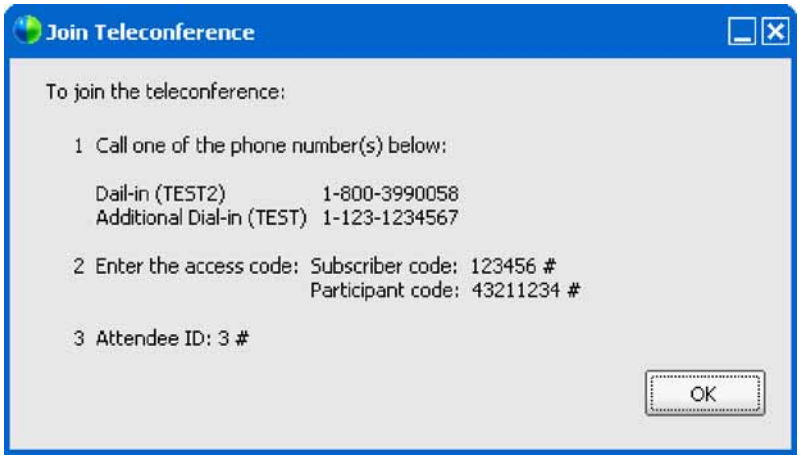
4.3.2 Join Teleconference

Once the meeting begins, the Join Teleconference screen displays the following teleconferencing information:

- Call in phone number.
- Meeting ID number.
- Attendee ID number.

If applicable, this screen prompts for call back.

Figure 4-2 • Join Teleconference Screen

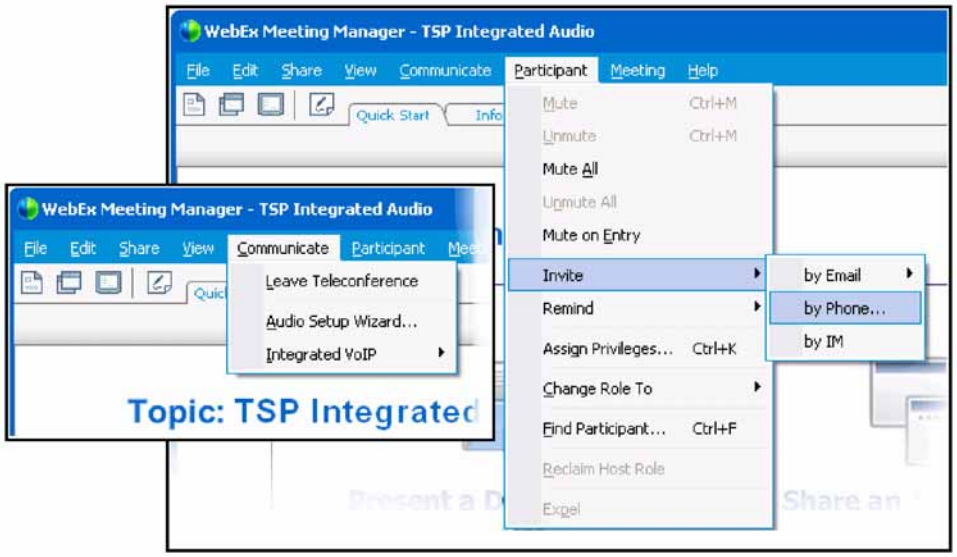


4.3.3 Participant Menu Selections

The following TSP teleconference call control functions may be invoked during the meeting:

- Leave teleconference.
- Mute.
- Unmute.
- Mute All.
- Unmute All.
- Phone another attendee to join the WebEx meeting.
- Expel.

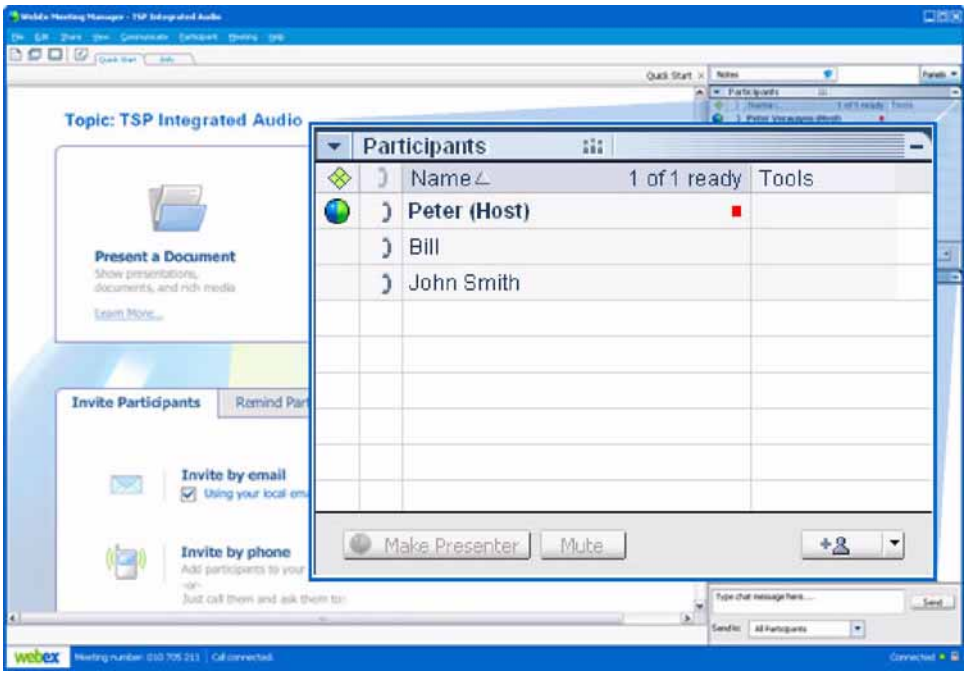
Figure 4-3 • Participant Menu Selections



4.3.4 Participant Icon Display

The **Meeting** window contains a section displaying each meeting participant with a phone icon indicating they are calling in through their teleconferencing service.

Figure 4-4 • Participant Icon Display in the Meeting Window



5

Integration Instructions with API Reference

This chapter provides API call sequence examples, the API reference, and a code example.

5.1 API Call Sequence Examples

5.1.1 User Call In

- Adapter sends the XML message *A2W_NotifyUserEnter* using HTTP post to WebEx using the *WbxHostName* associated with the conference. *Parameter ExtCallID* is null.
- WebEx returns a confirmation message *Confirmation* in the return HTTP post notifying the adapter the request was received.
- WebEx changes the Web user / meeting status to reflect that a user entered the event. WebEx checks the *ExtCallID* parameter in the *A2W_NotifyUserEnter*. If the parameter is null, and this is a call-in user, WebEx generates a new external call ID for it.
- WebEx sends *W2A_RspNotifyUserEnter* message to the adapter using the adapter host name given in the WebEx configuration file, and returns the new *External ID* and the original *MsgID* given in *A2W_NotifyUserEnter*.
- The adapter sends a confirmation message in the return HTTP post confirming the message was received and then associates the *External ID* with the *Bridge ID*.

5.1.2 User Call Out

- WebEx sends the XML message *W2A_Callout* to adapter using HTTP post to pass a valid external call ID.
- The adapter returns a confirmation message in the return HTTP post.
- The adapter works with the bridge trying to issue a call out.
- The adapter is notified by the bridge that the call out was successful, and sends an *A2W_RspCallOut* message to WebEx.
- WebEx returns a confirmation message.
- The adapter receives a notice from the bridge when the call-out user enters the conference. It then sends an *A2W_NotifyUserEnter* message with a valid *External Call ID* obtained from WebEx.
- WebEx returns a confirmation message.
- When WebEx finds parameter *ExtCallID* is valid, it knows this is a call-out user and there is no need to generate a new *External ID* for the user.
- WebEx sends a *W2A_RspNotifyUserEnter* message to the adapter with the valid *External Call ID* parameter.
- The adapter returns a confirmation message, and re-associates the *External Call ID* with the *Bridge Call ID* (re-associating is good practice).

5.2 Telephony API Reference for XML Implementation

5.2.1 Non-Command Elements

The following is a summary table of the non-command elements for the Telephony API.

Table 5-1 • Non-Command Elements Summary

Element Short	Description
`Confirmation`	This is a required return element for each HTTP transaction / request.
`TransID`	Required attribute for each HTTP transaction or request for each *WbxTSPSchema* element top level.

5.2.2 Confirmation

Confirmation is a required return element for each HTTP transaction or request. The receiving side (WebEx or adapter) should check all of the XML messages in the transaction against the XML definitions (there may be more than one XML message in a request). After all messages meet the requirements, an error code 0 (ok) is sent as a *Confirmation*. Otherwise, an appropriate error code is returned.

Attribute

```
Result - error code (0=ok)
```

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <Confirmation Result="x">
   </Confirmation>
</WbxTSPSchema>
```

5.2.3 TransID

TransID is a required attribute for each HTTP transaction or request for *WbxTSPSchema* element (i.e., top-level). It is a unique ID that identifies the HTTP transaction.

Note TransID is used to resend a request if, and only if, the request failed in previous trials.

Important The receiving side (adapter or WebEx) should always check each request *Transaction ID* to determine if the same request was received previously. If so, no other action should be taken except to return a confirmation. Under normal circumstances actions, requests with the same ID should not be responded to more than once by the receiving side.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
</WbxTSPSchema>
```

5.2.4 Command Elements

API Command Messages initiated from WebEx side have the prefix "W2A_" (WebEx to Adapter) and are supported by the adapter. Those from the adapter side have the prefix "A2W_" (Adapter to WebEx.) and are supported by WebEx.

The following is a summary table of the command elements for the Telephony API.

Table 5-2 • Command Elements Summary

Element	Short Description
W2A_AddToSubConference	WebEx server requests the partner bridge to move a set number of callers from the main conference to a sub-conference.
W2A_AuthenticateAccount	WebEx server requests to authenticate an account in the partner's conference system.
W2A_Callout	WebEx server requests the bridge to call a given number.
W2A_ChangeCallPrivilege	WebEx server requests the bridge to change the privilege of a call (e.g., muting or unmuting a call).
W2A_CloseConference	WebEx server requests to close a conference on the bridge.
W2A_CloseSubConference	WebEx server requests the bridge to end a sub-conference.
W2A_CreateAccount	WebEx server requests to create a telephony conference account in the partner's conference system.
W2A_CreateConference	WebEx server requests to create a conference on the bridge.
W2A_CreateSubConference	WebEx server requests to create a sub-conference on the bridge.
W2A_DeleteAccount	WebEx server requests to delete a telephony conference account from the partner's conference system.
W2A_DropCall	WebEx server requests the bridge to drop a call or to expel an attendee.
W2A_GetBridgeInfo	WebEx server asks for the type of service the bridge supports.
W2A_GetBridgeStatus	WebEx server queries the bridge adapter and the bridge's health status.
W2A_GetCallInfo	WebEx server requests the bridge to send information for a specified call.
W2A_GetConfInfo	WebEx server requests the bridge to send information for a specified conference.
W2A_GetResource	WebEx server asks for resources on the bridge (e.g., maximum number of lines).
W2A_Reset	WebEx server requests the adapter reset objects related to that WebEx server. This is sent whenever a WebEx server restarts.
W2A_RemoveFromSubConference	WebEx server requests the bridge to remove a set of callers from a sub-conference.
W2A_RspMessageError	WebEx server notifies the adapter that the request in a previously received API message from the adapter cannot be executed.

Table 5-2 • Command Elements Summary (Continued)

Element	Short Description
W2A_UpdateAccount	Webex server requests to update an existing account in the partner's conference system.
W2A_UpdateConference	WebEx server requests to change the status of a conference on the bridge.

Command Sub-elements

The following is a summary table of command sub-elements for the Telephony API.

Table 5-3 • Command Sub-elements Summary

Element	Sub-element	Short Description
W2A_Callout	PhoneNum	The Phone number to call.

5.2.5 W2A_AddToSubConference

WebEx server requests the partner bridge to move a set number of callers from the main conference to a sub-conference.

Attributes

Table 5-4 • W2A_AddToSubConference Attributes

Attribute	Required?	Description
ExtConfID	Yes	*External ID* of the conference on the WebEx server.
ExtCallID	Yes	The *ExtCallerID* to be added to sub-conference.
ExtSubConfID	Yes	*External ID* of the sub-conference.
ModeratorFlag	Yes	Flag (1/0) to determine whether the caller is a moderator.
MsgID	Yes	A unique ID of the message string.
WbxHostName	Yes	WebEx TSP server hostname and port number string.

Syntax for XML document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
<TransID>xxx</TransID>
   <W2A_AddToSubConference MsgID="xxx"
      WbxHostName="xxx" ExtConfID="xxx"
      ExtSubConfID="xxx" ExtCallID="xxx"
      ModeratorFlag="x">
   </W2A_AddToSubConference>
</WbxTSPSchema>
```

5.2.6 W2A_AuthenticateAccount

WebEx server requests to authenticate an account in the partner's conference system.

Attributes

Table 5-5 • W2A_AuthenticateAccount Attribute

Attribute	Required?	Description
AccessCode	No	Access code. This field is no longer used.
ExtUserID	No	An external ID of the user who requested the authentication.
MsgID	Yes	A unique ID of the message.
ParticipantPass	Yes	The code number that uniquely identifies the account of a participant.
SubscriberCode	Yes	The code number that provides access to subscriber control features.
TollFreeNum	No	The toll free phone number from which a subscriber or participant calls. You should use the character "-" to separate the country code from the other part of the number. For example, 1-8661234567.
TollNum	No	The toll phone number from which a subscriber or participant calls. You should use the character "-" to separate the country code from the other part of the number. For example, 1-4081234567.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_AuthenticateAccount
      MsgID="xxx" WbxHostName="xxx" TollFreeNum="xxx"
      TollNum="xxx" AccessCode="xxx" SubscriberCode="xxx"
      ParticipantPass="xxx" ExtUserID=”xxx”>
   </W2A_AuthenticateAccount>
</WbxTSPSchema>
```

5.2.7 W2A_Callout

WebEx server requests the bridge to call a given number. The adapter should use the corresponding response API to pass the return value, i.e., *A2W_RspCallout*.

Note The adapter should not call out after exceeding the maximum time-out period.

Attributes

Table 5-6 • W2A_Callout Attributes

Attribute	Required?	Description
ExtCallID	Yes	*External ID* of the call on the WebEx server side.
ExtConfID	Yes	External conference ID on the WebEx server side. **Note** This is a different ID from that in the adapter side.
JoinMode	No	The mode in which a participant joins the conference. Valid values are: ■ 0 - Talk with the host privately before joining the conference (Blast=FALSE). ■ 1 - Join the conference by pressing "1". ■ 2 - Join the conference without pressing "1".
MaxWaitingTime	Yes	WebEx maximum waiting time in seconds before issuing a time-out.
MsgID	Yes	A unique ID of the message.
ParticipantType	No	Optional. An integer indicating the type of the participant. Valid values are: ■ 1 - Subscriber. ■ 2 - Participant. ■ 3 - Operator. ■ 6 - Silent Monitor (Sales Center Only).
Privilege	Yes	Initial privilege in the conference. ■ 0 - Speak and listen. ■ 1 - Listen only.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Sub-element: PhoneNum

Table 5-7 • W2A_Callout Sub-element

Sub-element	Description
PhoneNum	The phone number to call.

Sub-element Syntax for XML Document

```
<PhoneNum>
   <CountryCode>xxx</CountryCode> (optional)
   <AreaCode>xxx</AreaCode>  (optional)
   <LocalNumber>xxx</LocalNumber>
   <Extension>xxx</Extension> (optional)
</PhoneNum>
```

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_Callout
      MsgID="xxx" WbxHostName="xxx" ExtConfID="xxx"
      ExtCallID="xxx" Privilege="x" MaxWaitingTime="xxx"
      ParticipantType="x" JoinMode="x">
      <PhoneNum>
         <CountryCode>xxx</CountryCode>
         <AreaCode>xxx</AreaCode>
         <LocalNumber>xxx</LocalNumber>
         <Extension>xxx</Extension>
      </PhoneNum>
   </W2A_Callout >
</WbxTSPSchema>
```

5.2.8 W2A_ChangeCallPrivilege

WebEx server requests the bridge to change the privilege of a call (e.g., muting or unmuting a call). The adapter should use the corresponding response API to pass the return value, i.e., *A2W_RspChangeCallPrivilege*.

Attributes

Table 5-8 • W2A_ChangeCallPrivilege Attributes

Attribute	Required?	Description
ExtCallID	Yes	*External ID* of the call on the WebEx server side.
ExtConfID	Yes	External conference ID on the WebEx server side. **Note** This is a different ID from that in the adapter side.
MsgID	Yes	A unique ID of the message.
Privilege	Yes	The new privilege Privilege: ■ 0 - Speak and listen. ■ 1 - Listen only.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_ChangeCallPrivilege
      MsgID="xxx" WbxHostName="xxx" ExtConfID="xxx"
      ExtCallID="xxx" Privilege="x">
   </W2A_ChangeCallPrivilege>
</WbxTSPSchema>
```

5.2.9 W2A_CloseConference

WebEx server requests to close a conference on the bridge. The adapter should use the corresponding response API to pass the return value, i.e., *A2W_RspCloseConference*.

Attributes

Table 5-9 • W2A_CloseConference Attributes

Attribute	Required?	Description
ExtConfID	Yes	External conference ID on the WebEx server side. **Note** This is a different ID from that in the adapter side.
KeepAlive	No	A boolean variable that controls whether or not the teleconference should end. ■ 0 = Close the teleconference. ■ 1 = Keep the teleconference running.
MsgID	Yes	A unique ID of the message.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
      <W2A_CloseConference
         MsgID="xxx" WbxHostName="xxx" ExtConfID="xxx" KeepAlive=”1”>
   </W2A_CloseConference>
</WbxTSPSchema>
```

5.2.10 W2A_CloseSubConference

WebEx server requests the bridge to end a sub-conference.

Attributes

Table 5-10 • W2A_CloseSubConference Attributes

Attribute	Required?	Description
ExtConfID	Yes	External ID of the conference on the WebEx server.
ExtSubConfID	Yes	External ID of the sub-conference.
MsgID	Yes	A unique ID of the message string.
WbxHostName	Yes	WebEx TSP server hostname and port number string.

Syntax for XML document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_CloseSubConference
      MsgID="xxx" WbxHostName="xxx" ExtConfID="xxx"
      ExtSubConfID="xxx">
   </W2A_CloseSubConference>
</WbxTSPSchema>
```

5.2.11 W2A_CreateAccount

WebEx server requests to create a telephony conference account in the partner's conference system. As this is an optional feature, if the adapter does not support this feature, it should return the error code (-20).

Attributes

Table 5-11 • W2A_CreateAccount Attribute

Attribute	Required?	Description
AccessCode	No	Access code. This field is no longer used.
ExtUserID	No	An external ID of the user who created the account.
MaxNumAttendee	Yes	The maximum number of attendees allowed in a conference.
MsgID	Yes	A unique ID of the message.
ParticipantPass	Yes	The code number that uniquely identifies the account of a participant.
RollCall	Yes	An integer specifying whether a tone is required when a participant joins the conference. Valid values are: ■ 0 - Tone required. ■ 1 - Named entry. ■ 2 - No tone required (Silence).
SubscriberCode	Yes	The code number that provides access to subscriber control features.

Table 5-11 • W2A_CreateAccount Attribute (Continued)

Attribute	Required?	Description
TollFreeNum	No	The toll free phone number from which a subscriber or participant calls. Use the character "-" to separate the country code from the other parts of the number. For example, 1-8661234567.
TollNum	No	The toll phone number from which a subscriber or participant calls. Use the character "-" to separate the country code from the other parts of the number. For example, 1-4081234567.
WbxHostID	No	WebEx host ID.
WbxHostName	Yes	WebEx TSP server host name and port number string.
WbxSiteID	No	WebEx Site ID.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_CreateAccount
      MsgID="xsx" WbxHostName="xxx" TollFreeNum="xxx"
      TollNum="xxx" AccessCode="xsx" SubscriberCode="xxx"
      ParticipantPass="xxx" MaxNumAttendee="xxx" RollCall="xxx"
      WbxHostID="xxx" WbxSiteID="xxx">
   </W2A_CreateAccount>
</WbxTSPSchema>
```

5.2.12 W2A_CreateConference

WebEx server requests to create a conference on the bridge. When attendees enter the conference key from the keypad; the adapter, or bridge, should place them together in the same conference if they have the same conference keys.

The adapter should use the corresponding response API to pass the result, i.e., *A2W_RspCreateConference*.

Note The WebEx telephony server's host name is passed to the adapter side. Any future API messages from the adapter to WebEx should be sent to the host if the messages are related to this conference.

Attributes

Table 5-12 • W2A_CreateConference Attributes

Attribute	Required?	Description
AccessCode	No	Access code. No longer used.
AnonymousID	No	The AttendeeID that will represent a silent-monitor. This attribute is added for security purposes, giving the adapter / bridge the ability to verify that a silent-joiner is genuine, and not an eavesdropper.
ConfKey	No	Conference key, based on which participants are placed into an audio conference.
ExtConfID	Yes	External conference ID on the WebEx server side. **Note** This is a different ID from that in the adapter side.
ExtUserID	No	An external ID of the user who created the account.
MaxNumAttendee	Yes	Maximum number of attendees in the conference.
MsgID	Yes	A unique ID of the message.
NBRCallID	No	An integer representing the AttendeeID reserved for the NBR call leg.
ParticipantPass	No	The code number that uniquely identifies the account of a participant, for static accounts to create a conference.
Selection	No	A string representing the following selections in the order from left to right: ■ First character: *RollCall*. Specifies whether a tone is required when a participant joins the conference. ❏ 0 - Tone required. ❏ 1 - Named entry. ❏ 2 - No tone required (Silence). ■ Second character: *ContactOperator*. Specifies whether contacting the operator is allowed. Valid values are: ❏ 0 - Not allowed. ❏ 1 - Allowed. ■ Third character: *TollFree*. Specifies whether the toll free number feature is enabled. Valid values are: ❏ 0 - Not enabled. ❏ 1 - Enabled.
SubscriberCode	No	The code number that provides access to subscriber control features, for static accounts to create a conference.
TollFreeNum	No	The toll free phone number from which a subscriber or participant calls, for static accounts to create a conference. You should use the character "-" to separate the country code from the other part of the number. For example, 1-4081234567.

Table 5-12 • W2A_CreateConference Attributes (Continued)

Attribute	Required?	Description
TollNum	No	The toll phone number from which a subscriber or participant calls, for static accounts to create a conference. You should use the character "-" to separate the country code from the other part of the number. For example, 1-8661234567.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_CreateConference
      MsgID="xxx" WbxHostName="xxx" ExtConfID="xxx"
      ConfKey="xxx" MaxNumAttendee="xxx" Selection="xxx"
      TollFreeNum="xxx" TollNum="xxx" AccessCode="xxx"
      SubscriberCode="xxx" ParticipantPass="xxx">
   </W2A_CreateConference>
</WbxTSPSchema>
```

5.2.13 W2A_CreateSubConference

WebEx server requests to create a sub-conference on the bridge.

Attributes

Table 5-13 • W2A_CreateSubConference Attributes

Attribute	Required?	Description
ExtConfID	Yes	External ID of the conference on the WebEx server.
ExtSubConfID	Yes	External ID of the sub-conference.
MsgID	Yes	A unique ID of the message string.
WbxHostName	Yes	WebEx TSP server hostname and port number string.

Syntax For XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_CreateSubConference
      MsgID="xxx" WbxHostName="xxx" ExtConfID="xxx"
      ExtSubConfID="xxx">
   </W2A_CreateSubConference>
</WbxTSPSchema>
```

5.2.14 W2A_DeleteAccount

WebEx server requests to delete a telephony conference account from the partner's conference system. Since this is an optional feature, if the adapter does not support this feature, it returns the error code (-20).

Attributes

Table 5-14 • W2A_DeleteAccount Attributes

Attribute	Required?	Description
AccessCode	No	The code number that uniquely identifies the account of a subscriber.
MsgID	Yes	A unique ID of the message.
ParticipantPass	No	The code number that uniquely identifies the account of a participant.
SubscriberCode	No	The code number that provides access to subscriber control features.
TollFreeNum	No	The toll free phone number from which a subscriber or participant calls. You should use the character "-" to separate the country code from the other part of the number. For example, 1-8661234567.
TollNum	No	The toll phone number from which a subscriber or participant calls. You should use the character "-" to separate the country code from the other part of the number. For example, 1-4081234567.
WbxHostName	Yes	WebEx TSP server host name and port number string.
WbxSiteID	No	The WebEx site ID.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_DeleteAccount
      MsgID="xxx" WbxHostName="xxx" TollFreeNum="xxx"
      TollNum="xxx" AccessCode="xxx" SubscriberCode="xxx"
      ParticipantPass="xxx">
   </W2A_DeleteAccount>
</WbxTSPSchema>
```

5.2.15 W2A_DropCall

WebEx server requests the bridge to drop a call or expel an attendee. The adapter should use corresponding response API to pass the return value, i.e., *A2W_RspDropCall*.

Attributes

Table 5-15 • W2A_DropCall Attributes

Attribute	Required?	Description
ExtCallID	Yes	*External ID* of the call on the WebEx server side.
ExtConfID	Yes	External conference ID on the WebEx server side. **Note** This is a different ID from that in the adapter side.
MsgID	Yes	A unique ID of the message.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_DropCall
      MsgID="xxx" WbxHostName="xxx" ExtConfID="xxx"
      ExtCallID="xxx">
   </W2A_DropCall>
</WbxTSPSchema>
```

5.2.16 W2A_GetBridgeInfo

WebEx server asks for the type of service the bridge supports. The adapter uses the corresponding response API to pass the return value, i.e., *A2W_RspGetBridgeInfo*.

Attributes

Table 5-16 • W2A_GetBridgeInfo Attributes

Attribute	Required?	Description
MsgID	Yes	A unique ID of the message.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_GetBridgeInfo
      MsgID="xxx" WbxHostName="xxx">
   </W2A_GetBridgeInfo>
</WbxTSPSchema>
```

5.2.17 W2A_GetBridgeStatus

WebEx server queries the bridge adapter and the bridge's health status. This API could be called frequently from the WebEx server side (up to once every 30 seconds). The adapter should handle this call efficiently.

The adapter should use the corresponding response API to pass the return value, i.e., *A2W_RspGetBridgeStatus*.

Attributes

Table 5-17 • W2A_GetBridgeStatus Attributes

Attribute	Required?	Description
MsgID	Yes	A unique ID of the message.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_GetBridgeStatus
      MsgID="xxx" WbxHostName="xxx">
   </W2A_GetBridgeStatus>
</WbxTSPSchema>
```

5.2.18 W2A_GetCallInfo

WebEx server requests the bridge to send information for a specified call. The adapter should use the corresponding response API to pass the return value, i.e., *A2W_RspGetCallInfo*.

Attributes

Table 5-18 • W2A_GetCallInfo Attributes

Attribute	Required?	Description
ExtCallID	Yes	External ID of the call on the WebEx server side.
ExtConfID	Yes	External conference ID on the WebEx server side. **Note** This is a different ID from that in the adapter side.
MsgID	Yes	A unique ID of the message.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_GetCallInfo
      MsgID="xxx" WbxHostName="xxx" ExtConfID="xxx"
      ExtCallID="xxx">
   </W2A_GetCallInfo>
</WbxTSPSchema>
```

5.2.19 W2A_GetConfInfo

WebEx server requests the bridge to send information for a specified conference. The adapter should use the corresponding response API to pass the return value, i.e., *A2W_RspGetConfInfo*.

Attributes

Table 5-19 • W2A_GetConfInfo Attributes

Attribute	Required?	Description
ExtConfID	Yes	External ID of the conference on the WebEx server side.
MsgID	Yes	A unique ID of the message.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_GetConfInfo
      MsgID="xxx" WbxHostName="xxx" ExtConfID="xxx">
   </W2A_GetConfInfo>
</WbxTSPSchema>
```

5.2.20 W2A_GetResource

WebEx server asks for resources on the bridge (i.e., maximum number of lines). The adapter should use the corresponding response API to pass the return value, i.e., *A2W_RspGetResource*.

Attributes

Table 5-20 • W2A_GetResource Attributes

Attribute	Required?	Description
MsgID	Yes	A unique ID of the message.
ResourceType	Yes	■ 0 - Maximum number of lines on the bridge. ■ 1 - Number of lines available now. ■ 2 - Maximum number of participants in one conference. ■ 3 - Maximum number of participants in one conference that can both speak and listen. ■ 4 - Number of participants that can both speak and listen in the same meeting as each new meeting begins.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_GetResource
      MsgID="xxx" WbxHostName="xxx" ResourceType="x">
   </W2A_GetResource>
</WbxTSPSchema>
```

5.2.21 W2A_RspMessageError

WebEx server notifies the adapter that the request in a previously received API message from the adapter cannot be executed. For example:

The API message is not recognizable.

This command element has no expected adapter response element.

Attributes

Table 5-21 • W2A_RspMessageError Attributes

Attribute	Required?	Description
MsgID	Yes	The original ID of the message.
MsgName	Yes	The API message name.
Result	Yes	ErrorCode.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_RspMessageError
      MsgID="xxx" MsgName ="xxx" Result ="xxx">
   </W2A_RspMessageError>
</WbxTSPSchema>
```

5.2.22 W2A_Reset

WebEx server requests the adapter reset objects related to that WebEx server. This is sent whenever a WebEx server restarts. Upon receiving this message, the adapter should flag all conferences requested from the WebEx server with the given host names or URLs. The previous meetings should be removed once completed.

The adapter should use the corresponding response API to pass the return value, i.e., *A2W_RspReset*.

Important For each conference, the adapter should wait until all attendees in the meeting leave, then close the conference. This is important if the WebEx server crashes for some reason and recovers.

Attributes

Table 5-22 • W2A_Reset Attributes

Attribute	Required?	Description
MsgID	Yes	A unique ID of the message. The adapter should include this ID in the response message to identify the requested response.
WbxHostName	Yes	WebEx TSP server host name and port number string (e.g., svr1.webex.com:80) The adapter should use this string to send a corresponding response element to the WebEx server.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_Reset
      MsgID="xxx" WbxHostName="xxx">
   </W2A_Reset>
</WbxTSPSchema>
```

5.2.23 W2A_RemoveFromSubConference

WebEx server requests the bridge to remove a set of callers from a sub-conference.

Attributes

Table 5-23 • W2A_RemoveFromSubConference Attributes

Attribute	Required?	Description
ExtCallID	Yes	The *ExtCallerID* to be removed from sub-conference.
ExtConfID	Yes	*External ID* of the conference on the WebEx server.
ExtSubConfID	Yes	*External ID* of the sub-conference.
MsgID	Yes	A unique ID of the message string.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_RemoveFromSubConference
      MsgID="xxx" WbxHostName="xxx" ExtConfID="xxx"
      ExtSubConfID="xxx" ExtCallID="xxx">
   </W2A_RemoveFromSubConference>
</WbxTSPSchema>
```

5.2.24 W2A_UpdateAccount

WebEx server requests to update an existing account in the partner's conference system. Since this is an optional feature, if the adapter does not support this feature, it should return an error code (-20).

Attributes

Table 5-24 • W2A_UpdateAccount Attributes

Attribute	Required?	Description
AccessCode_new	No	The new access code to which you want to set. See also *AccessCode_old*..
AccessCode_old	No	The old code number that uniquely identifies the account of a subscriber.
MaxNumAttendee_new	Yes	The new maximum number of attendees to which you want to set.
MsgID	Yes	A unique ID of the message.
ParticipantPass_new	Yes	The new participant password to which you want to set. See also *ParticipantPass_old*.

Table 5-24 • W2A_UpdateAccount Attributes (Continued)

Attribute	Required?	Description
ParticipantPass_old	Yes	The old code number that uniquely identifies the account of a participant.
RollCall_new	Yes	The new integer value that indicates whether a tone is required when a participant joins the conference. Valid values are: ■ 1 - Tone required. ■ 2 - Named entry. ■ 3 - No tone required (silence).
SubscriberCode_new	Yes	The new subscriber code to which you want to set. See also *SubscriberCode_old*.
SubscriberCode_old	Yes	The old code number that provides access to subscriber control features.
TollFreeNum_new	No	The new toll free phone number to which you want to set. Use the character "-" to separate the country code and other parts of the number. For example, 1-8661234567. See also *TollFreeNum_old*..
TollFreeNum_old	No	The old toll free phone number from which a subscriber or participant calls. You should use the character "-" to separate the country code from the other part of the number. For example, 1-8661234567.
TollNum_new	No	The new toll phone number to which you want to set. Use the character "-" to separate the country code and other parts of the number. For example, 1-4081234567. See also *TollNum_old*..
TollNum_old	No	The old toll phone number from which a subscriber or participant calls. Use the character "-" to separate the country code and other parts of the number. For example, 1-4081234567.
WbxHostName	Yes	WebEx TSP server host name and port number.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_UpdateAccount
      MsgID="xxx" WbxHostName="xxx" TollFreeNum_old="xxx"
      TollNum_old="xxx" AccessCode_old="xxx"
      SubscriberCode_old="xxx" ParticipantPass_old="xxx"
      TollFreeNum_new="xxx" TollNum_new="xxx"
      AccessCode_new="xxx" SubscriberCode_new="xxx"
      ParticipantPass_new="xxx" MaxNumAttendee_new="xxx"
      RollCall_new="xxx">
   </W2A_UpdateAccount>
</WbxTSPSchema>
```

5.2.25 W2A_UpdateConference

WebEx server requests to change the status of a conference on the bridge.

Attributes

Table 5-25 • W2A_UpdateConference Attributes

Attribute	Required?	Description
ActionCode	Yes	Requested action on the conference. Valid values are: ■ 1 - Lock the conference (i.e., do not allow a new attendee to join the conference). ■ 2 - Unlock the conference. ■ 3 - The caller with the *ExtCallID* requests to talk to the operator privately. ■ 4 - The caller with the *ExtCallID* requests to talk to the operator in the conference. ■ 5 - The caller with the *ExtCallID* requests the host role. ■ 6 - The participant comes back to the conference after talking to the operator. ■ 7 - Cancel the request to talk to the operator. ■ 8 - Turn on Active Speaker. ■ 9 - Turn off Active Speaker.
ExtCallID	Yes	A sequence number that identifies a calling user. This number is unique within a conference.
ExtConfID	Yes	External conference ID on the WebEx server side. **Note** This is a different ID from that in the adapter side.
MsgID	Yes	A unique ID of the message.
WbxHostName	Yes	WebEx TSP server host name and port number string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_UpdateConference
      MsgID="xxx" WbxHostName="xxx" ExtConfID="xxx"
      ActionCode="xxx" ExtCallID="xxx">
   </W2A_UpdateConference>
</WbxTSPSchema>
```

5.2.26 Expected Adapter Response Elements

The following is a summary table of expected adapter response elements for the Telephony API.

Table 5-26 • Expected Adapter Response Elements Summary

Element	Short Description
A2W_RspAddToSubConference	A message giving the result of *W2A_AddToSubConference.*
A2W_RspAuthenticateAccount	A message giving the result of *W2A_AuthenticateAccount.*
A2W_RspCallout	A message giving the result of *W2A_Callout*.
A2W_RspChangeCallPrivilege	A message giving the result of *W2A_ChangeCallPrivilege*.
A2W_RspCloseConference	A message giving the result of *W2A_CloseConference*.
A2W_RspCloseSubConference	A message giving the result of *W2A_CloseSubConference.*
A2W_RspCreateAccount	A message giving the result of *W2A_CreateAccount.*
A2W_RspCreateConference	A message giving the result of *W2A_CreateConference*.
A2W_RspCreateSubConference	A message giving the result of *W2A_CreateSubConference.*
A2W_RspDeleteAccount	A message giving the result of *W2A_DeleteAccount*.
A2W_RspDropCall	A message giving the result of *W2A_DropCall*.
A2W_RspGetBridgeInfo	A message giving the result of *W2A_GetBridgeInfo*.
A2W_RspGetBridgeStatus	A message giving the result of *W2A_GetBridgeStatus*.
A2W_RspGetCallInfo	A message giving the result of *W2A_GetCallInfo*.
A2W_RspGetConfInfo	A message giving the result of *W2A_GetConfInfo*.
A2W_RspGetResource	A message giving the result of *W2A_GetResource*.
A2W_RspMessageError	The adapter notifies WebEx that the request in a previously received API message from WebEx cannot be executed.
A2W_RspRemoveFromSubConference	A message giving the result of *W2A_RemoveFromSubConference.*
A2W_RspReset	Adapter should use *WbxHostName* as target server to send this response element to WebEx.
A2W_RspUpdateAccount	A message giving the result of *W2A_UpdateAccount*.
A2W_RspUpdateConference	A message giving the result of *W2A_UpdateConference*.

5.2.27 A2W_RspCloseSubConference

A message giving the result of *W2A_CloseSubConference*.

Attributes

Table 5-27 • A2W_RspCloseSubConference Attributes

Attribute	Required?	Description
MsgID	Yes	A unique ID of the message string.
Result	Yes	Error code (0 = success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspCloseSubConference
      MsgID="xxx" Result="x">
   </A2W_RspCloseSubConference>
</WbxTSPSchema>
```

5.2.28 A2W_RspAddToSubConference

A message giving the result of *W2A_AddToSubConference.*

Attributes

Table 5-28 • A2W_RspAddToSubConference Attributes

Attribute	Required?	Description
MsgID	Yes	A unique ID of the message string.
Result	Yes	Error Code (0=success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspAddToSubConference
      MsgID="xxx" Result="x">
   </A2W_RspAddToSubConference>
</WbxTSPSchema>
```

5.2.29 A2W_RspAuthenticateAccount

A message giving the result of *W2A_AuthenticateAccount*.

Attributes

Table 5-29 • A2W_RspAuthenticateAccount Attributes

Attribute	Required?	Description
MsgID	Yes	ID of the original message.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspAuthenticateAccount
      MsgID="xxx" Result="x">
   </A2W_RspAuthenticateAccount>
</WbxTSPSchema>
```

5.2.30 A2W_RspCallout

A message giving the result of *W2A_Callout*. The adapter uses this message to notify WebEx when a call out is successful or if it failed. If successful, a subsequent message, *A2W_NotifyUserEnter*, is sent to WebEx after the user joins the conference.

Attributes

Table 5-30 • A2W_RspCallout Attributes

Attribute	Required?	Description
MsgID	Yes	ID of the original message.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspCallout
      MsgID="xxx" Result="x">
   </A2W_RspCallout>
</WbxTSPSchema>
```

5.2.31 A2W_RspChangeCallPrivilege

A message giving the result of *W2A_ChangeCallPrivilege*.

Attributes

Table 5-31 • A2W_RspChangeCallPrivilege Attributes

Attribute	Required?	Description
MsgID	Yes	ID of the original message.
Privilege	No	The new privilege.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspChangeCallPrivilege
      MsgID="xxx" Privilege="x" Result="x">
   </A2W_RspChangeCallPrivilege>
</WbxTSPSchema>
```

5.2.32 A2W_RspCloseConference

A message giving the result of *W2A_RspCloseConference*.

Attributes

Table 5-32 • A2W_RspCloseConference Attributes

Attribute	Required?	Description
MsgID	Yes	ID of the original message.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspCloseConference
      MsgID="xxx" Result="x">
   </A2W_RspCloseConference>
</WbxTSPSchema>
```

5.2.33 A2W_RspCreateAccount

A message giving the result of *W2A_CreateAccount*.

Attributes

Table 5-33 • A2W_RspCreateAccount Attributes

Attribute	Required?	Description
MsgID	Yes	ID of the original message.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspCreateAccount
      MsgID="xxx" Result="x">
   </A2W_RspCreateAccount>
</WbxTSPSchema>
```

5.2.34 A2W_RspCreateConference

A message giving the result of *W2A_CreateConference*. This message is sent to WebEx to indicate the success or failure of creating a meeting. If successful, another message *A2W_NotifyConferenceChange* is sent to WebEx notifying that the conference is running and ready to be joined.

Attributes

Table 5-34 • A2W_RspCreateConference Attributes

Attribute	Required?	Description
MsgID	Yes	ID of the original message.
PSTNMerge	No	The DTMF code to be used on the bridge for merging the caller. Optional (example: *24).
Result	No	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspCreateConference
      MsgID="xxx" Result="x">
   </A2W_RspCreateConference>
</WbxTSPSchema>
```

5.2.35 A2W_RspCreateSubConference

A message giving the result of *W2A_CreateSubConference*.

Attributes

Table 5-35 • A2W_RspCreateSubConference Attributes

Attribute	Required?	Description
MsgID	Yes	A unique ID of the message string.
Result	Yes	Error Code (0=success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspCreateSubConference
      MsgID="xxx" Result="x">
   </A2W_RspCreateSubConference>
</WbxTSPSchema>
```

5.2.36 A2W_RspDeleteAccount

A message giving the result of *W2A_DeleteAccount*.

Attributes

Table 5-36 • A2W_RspDeleteAccount Attributes

Attribute	Required?	Description
MsgID	Yes	ID of the original message.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspDeleteAccount
      MsgID="xxx" Result="x">
   </A2W_RspDeleteAccount>
</WbxTSPSchema>
```

5.2.37 A2W_RspDropCall

A message giving the result of *W2A_DropCall*.

Attributes

Table 5-37 • A2W_RspDropCall Attributes

Attribute	Required?	Description
MsgID	Yes	ID of the original message.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspDropCall
      MsgID="xxx" Result="x">
   </A2W_RspDropCall>
</WbxTSPSchema>
```

5.2.38 A2W_RspGetBridgeInfo

A message giving the result of *W2A_GetBridgeInfo*.

Attributes

Table 5-38 • A2W_RspGetBridgeInfo Attributes

Attribute	Required?	Description
MsgID	Yes	ID of the original message.
Result	Yes	Error code (0=Success).
Type	Yes	Returned service type. Follow these definitions: ■ 1 - Conferencing. ■ 8 - Fax. ■ 16 - WebEx Remote Access authentication.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspGetBridgeInfo
      MsgID="xxx" Result="x" Type="xxx">
   </A2W_RspGetBridgeInfo>
</WbxTSPSchema>
```

5.2.39 A2W_RspGetBridgeStatus

A message giving the result of *W2A_GetBridgeStatus*.

Attributes

Table 5-39 • A2W_RspGetBridgeStatus Attributes

Attribute	Required?	Description
MsgID	Yes	Original Message ID.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspGetBridgeStatus
      MsgID="xxx" Result="x">
   </A2W_RspGetBridgeStatus>
</WbxTSPSchema>
```

5.2.40 A2W_RspGetCallInfo

A message giving the result of *W2A_GetCallInfo*.

Attributes

Table 5-40 • A2W_RspGetCallInfo Attributes

Attribute	Required?	Description
MsgID	Yes	ID of the original message.
Result	Yes	Error code (0=Success).
Status	Yes	Call status codes: ■ 0 - Normal. ■ 1 - Muted. ■ 2 - User leaves.
Type	Yes	Call type ■ 0 - call-in user. ■ 1 - call-out user.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspGetCallInfo
      MsgID="xxx" Result="x" Status="xxx" Type="xxx">
   </A2W_RspGetCallInfo>
</WbxTSPSchema>
```

5.2.41 A2W_RspGetConfInfo

A message giving the result of *W2A_GetConfInfo*.

Attributes

Table 5-41 • A2W_RspGetConfInfo Attributes

Attribute	Required?	Description
ConfStatus	Yes	Conference status.
MsgID	Yes	Original Message ID.
NumAttendee	Yes	Number of attendees in the conference.
NumCallBack	Yes	Number of callout users in the conference.
NumCallIn	Yes	Number of call-in users in the conference.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspGetConfInfo
      MsgID="xxx" Result="x" ConfStatus="x" NumAttendee="xxx"
      NumCallIn="xxx" NumCallBack="xxx">
   </A2W_RspGetConfInfo>
</WbxTSPSchema>
```

5.2.42 A2W_RspGetResource

A message giving the result of *W2A_GetResource*.

Attributes

Table 5-42 • A2W_RspGetResource Attributes

Attribute	Required?	Description
MsgID	Yes	ID of the Original message.
Resource	Yes	Resource of the corresponding resource type.
ResourceType	Yes	■ 0 - Maximum number of lines on the bridge. ■ 1 - Number of lines available now. ■ 2 - Maximum number of participants in one conference. ■ 3 - Maximum number of active participants in one conference. ■ 4 - Number of available active participants in one conference.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   < A2W_RspGetResource
      MsgID="xxx" ResourceType="x" Result="x" Resource="xxx">
   </A2W_RspGetResource>
</WbxTSPSchema>
```

5.2.43 A2W_RspMessageError

The adapter notifies WebEx that the request in a previously received API message from WebEx cannot be executed. For example:

> *The API message is not recognizable, or the specific function requested is not supported in this type of adapter/bridge.*

There is no expected response for this command element.

Attributes

Table 5-43 • A2W_RspMessageError Attributes

Attribute	Required?	Description
MsgID	Yes	The original ID of the message.
MsgName	Yes	The API message name.
Result	Yes	ErrorCode.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspMessageError
      MsgID="xxx" MsgName ="xxx" Result ="xxx">
   </A2W_RspMessageError>
</WbxTSPSchema>
```

5.2.44 A2W_RspRemoveFromSubConference

A message giving the result of *W2A_RemoveFromSubConference*.

Attributes

Table 5-44 • A2W_RspRemoveFromSubConference Attributes

Attribute	Required?	Description
MsgID	Yes	A unique ID of the message string.
Result	Yes	Error Code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspRemoveFromSubConference
      MsgID="xxx" Result="x">
   </A2W_RspRemoveFromSubConference>
</WbxTSPSchema>
```

5.2.45 A2W_RspReset

The adapter uses *WbxHostName* as a target server to send this response element to WebEx.

Attributes

Table 5-45 • A2W_RspReset Attributes

Attribute	Required?	Description
MsgID	Yes	The Message ID of the original message, *W2A_Reset*. WebEx uses this ID to associate this message with its original request.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspReset
      MsgID="xxx" Result="x">
   </A2W_RspReset>
</WbxTSPSchema>
```

5.2.46 A2W_RspUpdateAccount

A message giving the result of *W2A_UpdateAccount*.

Attributes

Table 5-46 • A2W_RspUpdateAccount Attributes

Attribute	Required?	Description
MsgID	Yes	The Message ID of the original message, *W2A_UpdateAccount*. WebEx will use this ID to associate this message with its original request.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspUpdateAccount
      MsgID="xxx" Result="x">
   </A2W_RspUpdateAccount>
</WbxTSPSchema>
```

5.2.47 A2W_RspUpdateConference

A message giving the result of *W2A_UpdateConference*.

Attributes

Table 5-47 • A2W_RspUpdateConference Attributes

Attribute	Required?	Description
MsgID	Yes	The message ID of the original message, *W2A_UpdateConference*. WebEx uses this ID to associate this message with its original request.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM Response" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_RspUpdateConference
      MsgID="xxx" Result="x">
   </A2W_RspUpdateConference>
</WbxTSPSchema>
```

5.3 API Messages Initiated from TSP Partner (or Adapter)

For the following calls, the adapter should use the TSP server host name associated with the corresponding conference to determine where the request is sent.

Command Elements

The following is a summary table of the command elements initiated by the TSP partner adapter to the WebEx Telephony API.

Table 5-48 • Command Elements Initiated from TSP Partner or Adapter Summary

Element	Short Description
A2W_CreateConnection	Starts the process of making a persistent connection to a WebEx TSP server.
A2W_NotifiyConferenceChange	The adapter notifies WebEx server that a conference's status has changed.
A2W_NotifyJoinSubConference	A function for the adapter to tell the TSP server that one caller is joining the sub-conference.
A2W_NotifySpeakingStatus	A message giving the ActiveSpeaker status to the TSP server.
A2W_NotifySubConfCallChange	A function for the adapter to tell the TSP server that sub-conference action result.
A2W_NotifySubConferenceChange	A function for the adapter to tell the TSP server the status of a sub-conference.
A2W_NotifyUserChange	The adapter notifies WebEx servers that a user's status has changed.
A2W_NotifyUserEnter	The adapter notifies WebEx servers that a user calls in.

5.3.1 A2W_CreateConnection

Adapter initiated command that starts the process of making a persistent connection to a WebEx TSP Server. This command is only used when a WebEx TSP server is running in OPL mode.

Attributes

Table 5-49 • A2W_CreateConnection Attributes

Attribute	Required?	Description
AdaURL	Yes	URL of the Adapter, whether it is inside of the DMZ or out.
MsgID	Yes	Unique message identifier.
samlResponse	Yes	The response from SAML used to authenticate the connection.

Table 5-49 • A2W_CreateConnection Attributes (Continued)

Attribute	Required?	Description
SiteName	Yes	The name of the WebEx site. Example: [sitename].webex.com.
webExID	Yes	The WebEx ID with Site Administrator Privileges.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
      <A2W_CreateConnection MsgID="xxx" webExID="xxx" SiteName="xxx"
      samlResponse="xxxxx" AdaURL="http://your.adapter.com">
   </A2W_CreateConnection>
</WbxTSPSchema>
```

5.3.2 A2W_NotifiyConferenceChange

The adapter notifies WebEx server that a conference's status has changed. No response is expected for this command element.

Attributes

Table 5-50 • A2W_NotifyConferenceChange Attributes

Attribute	Required?	Description
ExtConfID	Yes	External ID of the conference on the WebEx server side.
MsgID	Yes	A unique ID of the message.
Status	Yes	New conference status: ■ 0 - Running. ■ 1 - Conference is being created but not ready for joining yet. ■ 2 - Conference is locked. ■ 3 - Conference has ended.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_NotifyConferenceChange
      MsgID="xxx" ExtConfID="xxx" Status="xxx">
   </A2W_NotifyConferenceChange>
</WbxTSPSchema>
```

5.3.3 A2W_NotifyUserChange

The adapter notifies WebEx server that a user's status has changed. No response is expected for this command element.

Attributes

Table 5-51 • A2W_NotifyUserChange Attributes

Attribute	Required?	Description
ActionCode	No	An integer indicating the requested action. Valid values are: ■ 1 - No action. ■ 2 - Request for the participant to talk to the host privately. ■ 3 - Request to come back to the conference. ■ 4 - Request for talking to the operator privately. ■ 5 - Request for the operator to join the conference. ■ 6 - The request for talking to the operator privately is answered. ■ 7 - The request for the operator to join the conference is answered. ■ 8 - Merge caller with participant list entry.
ActionResult	No	The result of the requested action. Valid values are: ■ 0 - OK. ■ 1 - General error. ■ 2 - No operator.
AttendeeID	No	Only required if ActionCode=8. This is the WebEx meeting attendee ID the user enters in the phone keypad after the partner defined DTMF code.
DID	No	The DID number.
ExtCallID	Yes	*External ID* of the call on the WebEx server side.
ExtConfID	Yes	*External ID* of the conference on the WebEx server side.
MsgID	Yes	A unique ID of the message.
ParticipantType	No	Optional. An integer indicating the type of the participant. Valid values are: ■ 1 - Subscriber. ■ 2 - Participant. ■ 3 - Operator.
Status	Yes	New user status: ■ 0 - Normal. ■ 1 - Muted. ■ 2 - User leaves.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_NotifyUserChange
      MsgID="xxx" ExtConfID="xxx" ExtCallID ="xxx"
      Status="xxx" ParticipantType="x" DID="xxx" ActionCode="x"
      ActionResult="x">
   </A2W_NotifyUserChange>
</WbxTSPSchema>
```

5.3.4 A2W_NotifyUserEnter

The adapter notifies WebEx server that a user calls in.

Note The parameters for call-in and call-out users are different. WebEx gives a different response for each.

Attributes

Table 5-52 • A2W_NotifyUserEnter Attributes

Attribute	Required?	Description
ActionCode	No	Optional. An integer indicating the requested action. Valid values are: ■ 2 - The participant requests to talk to the host privately. ■ 7 - The request for the operator to join the conference is answered.
ActionResult	No	Optional. An integer indicating the result of the requested action. Valid values are: ■ 0 - OK. ■ 1 - General error. ■ 2 - No operator.
AttendeeID	No	For call-in user, this is the WebEx meeting attendee ID the user enters from the keypad. For call-out, this is null.
DID	No	Optional. The DID number.
ExtCallID	Yes	*Call ID* on the WebEx side. For call-in user, this value is null. For call-out, the adapter provides this value.
ExtConfID	Yes	*External ID* of the conference on the WebEx server side.

Table 5-52 • A2W_NotifyUserEnter Attributes (Continued)

Attribute	Required?	Description
MsgID	Yes	A unique ID of the message.
ParticipantType	No	Optional. An integer indicating the type of the participant. Valid values are: ■ 1 - Subscriber. ■ 2 - Participant. ■ 3 - Operator. ■ 6 - Silent Monitor (Sales Center Only).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_NotifyUserEnter
      MsgID="xxx" ExtConfID="xxx" ExtCallID="xxx"
      AttendeeID="xxx" ParticipantType="x" DID="xxx"
      ActionCode="x" ActionResult="x">
   </A2W_NotifyUserEnter>
</WbxTSPSchema>
```

5.3.5 A2W_NotifyJoinSubConference

A function for the adapter to tell the TSP server that one caller is joining the sub-conference.

Attributes

Table 5-53 • A2W_NotifyJoinSubConference Attributes

Attribute	Required?	Description
ExtCallID	Yes	The *ExtCallerID* to be added to sub-conference.
ExtConfID	Yes	External ID of the conference on the WebEx server.
ExtSubconfID	Yes	External ID of the sub-conference.
ModeratorFlag	Yes	Flag (1/0) to determine whether the caller is a moderator.
MsgID	Yes	A unique ID of the message string.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_NotifyJoinSubConference
      MsgID="xxx" ExtConfID="xxx"
      ExtSubConfID="xxx" ExtCallID="xxx"
```

```
      ModeratorFlag="x">
   </A2W_NotifyJoinSubConference>
</WbxTSPSchema>
```

Note This XML API is only used in the event that a TSP server goes down and the conference is fails-over to a new TSP server.

5.3.6 A2W_NotifySpeakingStatus

A message giving the *ActiveSpeaker* status to the TSP server.

Attributes

Table 5-54 • A2W_NotifySpeakingStatus Attributes

Attribute	Required?	Description
ExtCallIDs	Yes	List of *ExtCallerID*'s whose speaking status has changed. The *ExtCallIDs* are separately by "%".
ExtConfID	Yes	*External ID* of the conference on the WebEx server.
MsgID	Yes	A unique ID of the message.
SpeakingFlags	Yes	Flags (Y/N) to determine whether the caller is speaking or not. Each flag is separated by '%'. I.e. "N%Y%Y%N".

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_NotifySpeakingStatus
      MsgID="xxx" ExtConfID="xxx" ExtCallIDs="xxx"
      SpeakingFlags="xxx">
   </A2W_NotifySpeakingStatus>
</WbxTSPSchema>
```

5.3.7 A2W_NotifySubConfCallChange

A function for the adapter to tell the TSP server that a sub-conference action results.

Attributes

Table 5-55 • A2W_NotifySubConfCallChange Attributes

Attribute	Required?	Description
`ActionFlag`	Yes	Flag (1/0) ■ Add the *CallLeg* into *Subconf.* ■ Remove the *CallLeg* from *SubConf.*
`ExtCallID`	Yes	The *ExtCallID* to be added/removed in the sub-conference.
`ExtConfID`	Yes	*External ID* of the conference on the WebEx server.
`ExtSubConfID`	Yes	*External ID* of the sub-conference.
`MsgID`	Yes	A unique ID of the message string.
`Status`	Yes	The result of the action, 0 is success, -1 is general error.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_NotifySubConfCallChange
      MsgID="xxx" ExtConfID="xxx"
      ExtSubConfID="xxx" ExtCallID="xxx"
      ActionFlag="xxx" Status="xxx">
   </A2W_NotifySubConfCallChange>
</WbxTSPSchema>
```

Note This XML API is only used in the event that a TSP server goes down after a conference fails-over to a new TSP server.

5.3.8 A2W_NotifySubConferenceChange

A function for the adapter to tell the TSP server of the status of a sub-conference.

Attributes

Table 5-56 • A2W_NotifySubConferenceChange Attributes

Attribute	Required?	Description
ExtConfID	Yes	*External ID* of the conference of the WebEx server.
ExtSubConfID	Yes	*External ID* of the sub-conference.
MsgID	Yes	A unique ID of the message string.
Status	Yes	Status codes: ■ 0 - running. ■ 1 - end. ■ 2 - failover.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <A2W_NotifySubConferenceChange MsgID="xxx" ExtConfID="xxx"
      ExtSubConfID="xxx" Status="xxx">
   </A2W_NotifySubConferenceChange>
</WbxTSPSchema>
```

5.3.9 Expected WebEx Server Response Elements

The following is a summary table of the expected WebEx Telephony server response elements initiated from the WebEx Telephony server.

Table 5-57 • Expected WebEx Server Response Element Summary

Element	Short Description
W2A_RspCreateConnection	This response element confirms a persistent connection.
W2A_RspNotifyUserChange	This response element confirms W2A_NotifyUserChange completed successfully.
W2A_RspNotifyUserEnter	This response element always return a valid external call ID.

5.3.10 W2A_RspCreateConnection

This response element is used by the WebEx TSP Server to confirm that a persistent connection from the adapter is present.

Attributes

Table 5-58 • W2A_RspCreateConnection Attributes

Attribute	Required?	Description
MsgID	Yes	The unique message identifier.
Result	Yes	A code indicating success or failure. 0 = Success.

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_RspCreateConnection MsgID="xxx" Result="x">
   </W2A_RspCreateConnection>
</WbxTSPSchema>
```

5.3.11 W2A_RspNotifyUserChange

This response element is used by the WebEx TSP Server to confirm that the operation made to the user in *W2A_NotifyUserChange* was completed successfully.

Attributes

Table 5-59 • W2A_RspNotifyUserChange Attributes

Attribute	Required?	Description
AttendeeID	No	For a call-in user, this is the WebEx meeting attendee ID the user enters from the keypad.
ExtCallID	Yes	External ID of the call-leg on the WebEx server side.
ExtConfID	Yes	External ID of the conference on the WebEx server side.
MsgID	Yes	Message Unique Identifier.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_RspNotifyUserChange MsgID="xxx" ExtCallID="xxx" Result="x"
   ExtConfID="xxx" AttendeeID="xxx"></W2A_RspNotifyUserChange>
</WbxTSPSchema>
```

5.3.12 W2A_RspNotifyUserEnter

This response element always return a valid external call ID. If the *ExtCallID* passed-in is not null, it just returns that value. Otherwise, it generates an *External ID* and returns it to the adapter. The adapter should associate the *External ID* to the bridge *Internal ID* after receiving this response.

Attributes

Table 5-60 • W2A_RspNotifyUserEnter Attributes

Attribute	Required?	Description
AttendeeID	No	*Attendee ID* from the original message (optional).
ExtCallID	Yes	*Call ID* on the WebEx side. Should not be null.
ExtConfID	Yes	*External ID* of the conference on the WebEx server side (optional).
MsgID	Yes	Original message ID of *A2W_NotifyUserEnter*. The returned external ID can be associated with the caller through the message ID.
Result	Yes	Error code (0=Success).

Syntax for XML Document

```
<WbxTSPSchema Name="WTSPDOM" Version="1.0">
   <TransID>xxx</TransID>
   <W2A_RspNotifyUserEnter
```

```
        MsgID="xxx" ExtCallID="xxx" Result="x" ExtConfID=”xxx”>
    </W2A_RspNotifyUserEnter>
</WbxTSPSchema>
```

A

Code Definitions

A.1 Status Codes

A.1.1 Conference Status Codes

Codes used to indicate the status of the current conference:

- 0 = Running.
- 1 = Conference is being created but not ready for joining.
- 2 = Conference is locked (no more attendees may join).
- 3 = Conference has ended.

A.1.2 Call/User Status Codes

Codes used to indicate the status of a user:

- 0 = Normal.
- 1 = Muted.
- 2 = User leaves.

A.1.3 Call Privilege Codes

Codes used to indicate what privileges a user has in a given session:

- 0 = Talk and listen.
- 1 = Listen only.

A.2 Error Codes

Note Applies to both WebEx and the adapter side.

Table A-1 • Error Codes

Code	Description
0	Ok.
-1	General failure.
-2	No resource available.
-3	Cannot perform action requested because given conference already exists.
-4	Cannot perform action requested because given conference does not exist.
-5	Cannot perform action requested because given call/user already exists..
-6	Cannot perform action requested because given call/user does not exists.
-7	Invalid call-out number.
-8	Adapter gets time-out when talks to bridge.
-9	Unknown code received.
-10	Bridge error.
-11	Incorrect XML message received.
-12	Unrecognized new XML message received.
-13	Requested function is not supported.
-14	Incorrect password.
-15	Subscriber does not exist.
-16	AccessCode not accepted or duplicate.
-17	ParticipantPass not accepted or duplicate.
-18	Incorrect phone number.
-19	Bridge exceeds its capacity. New account cannot be created.
-20	Not supported by the TSP system.

B

Data Definition Agreements

B.1 Maximum Number of Messages in A Single Transaction

A transaction can contain more than one message. To make handling the transactions easier, both the WebEx and the adapter limit the maximum number of messages in a single transaction to 50. This means the XML document for a transaction can contain up to 51 elements under the root (50 messages and a *TranID* element).

B.2 Transaction ID And Message ID Generations

Transaction ID and *Message ID* are defined as an 11 digit, unique number sequence. WebEx recommends that TSP Adapters set the first five digits as the Julian Date of the day, and the right six digits from a sequence.

Note For details about Julian date, please refer to the relevant information at http://webexhibits.org/calendars/calendar-christian.html or http://www.maa.mhn.de/Scholar/times.html.

The following is an example:

- The Julian date of 8/05/2002 is: 2452491.
- If the next value of the *Message ID* sequence is 123, the next *Message ID* will be 52491000123.
  - Similarly, if the next value of the *Transaction ID* sequence is 1201, the next *Transaction ID* will be 52491001201.

Note The WebEx TSP server uses a different algorithm to generate the Transaction/Message IDs.

- TSP unique ID (2 digits)
- Tick of the day in seconds from 00:00:00 (5 digits)
- Sequence number: 0-9999 (4 digits)

B.3 Schema and Document Type Definition (DTD)

The TSP API supports two methods for describing the structure of XML documents. The first version of TSP utilized the Document Type Definition (DTD) specification which is considered to be out dated by most developers. WebEx currently maintains this format for backward compatibility. An example is included in this section.

The more modern method of describing TSP XML documents uses a schema (xsd). An example is included in this section.

B.3.1 DTD

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT WbxTSPSchema (TransID, (Confirmation)?, (W2A_Reset |
A2W_RspReset | W2A_GetBridgeStatus | A2W_RspGetBridgeStatus |
W2A_GetResource | A2W_RspGetResource | W2A_GetBridgeInfo |
A2W_RspGetBridgeInfo | W2A_CreateConference | A2W_RspCreateConference |
W2A_CloseConference | A2W_RspCloseConference | W2A_Callout |
A2W_RspCallout | W2A_DropCall | A2W_RspDropCall |
W2A_ChangeCallPrivilege | A2W_RspChangeCallPrivilege | W2A_GetConfInfo |
A2W_RspGetConfInfo | W2A_GetCallInfo | A2W_RspGetCallInfo |
W2A_UpdateConference | A2W_RspUpdateConference | W2A_RspMessageError |
W2A_AuthenticateAccount | A2W_RspAuthenticateAccount | W2A_CreateAccount
| A2W_RspCreateAccount | W2A_UpdateAccount | A2W_RspUpdateAccount |
W2A_DeleteAccount | A2W_RspDeleteAccount | A2W_NotifyUserEnter |
W2A_RspNotifyUserEnter | A2W_NotifyUserChange |
A2W_NotifyConferenceChange | A2W_RspMessageError |
A2W_NotifySpeakingStatus | W2A_CreateSubConference |
A2W_RspCreateSubConference | W2A_CloseSubConference |
A2W_RspCloseSubConference | W2A_AddToSubConference |
A2W_RspAddToSubConference | W2A_RemoveFromSubConference |
A2W_RspRemoveFromSubConference | A2W_NotifySubConferenceChange |
A2W_NotifyJoinSubConference | A2W_NotifySubConfCallChange |
A2W_CreateConnection | W2A_RspCreateConnection |
W2A_RspNotifyUserChange)*)>
<!ATTLIST WbxTSPSchema
   Name CDATA #REQUIRED
   Version CDATA #REQUIRED
>
<!ELEMENT TransID (#PCDATA)>
<!ELEMENT Confirmation (#PCDATA)>
```

```
<!ATTLIST Confirmation
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_Reset (#PCDATA)>
<!ATTLIST W2A_Reset
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
>
<!ELEMENT A2W_RspReset (#PCDATA)>
<!ATTLIST A2W_RspReset
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_GetBridgeStatus (#PCDATA)>
<!ATTLIST W2A_GetBridgeStatus
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
>
<!ELEMENT A2W_RspGetBridgeStatus (#PCDATA)>
<!ATTLIST A2W_RspGetBridgeStatus
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_GetResource (#PCDATA)>
<!ATTLIST W2A_GetResource
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   ResourceType CDATA #REQUIRED
>
<!ELEMENT A2W_RspGetResource (#PCDATA)>
<!ATTLIST A2W_RspGetResource
   MsgID CDATA #REQUIRED
   ResourceType CDATA #REQUIRED
   Resource CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_GetBridgeInfo (#PCDATA)>
<!ATTLIST W2A_GetBridgeInfo
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
>
<!ELEMENT A2W_RspGetBridgeInfo (#PCDATA)>
<!ATTLIST A2W_RspGetBridgeInfo
   MsgID CDATA #REQUIRED
```

```
   Result CDATA #REQUIRED
   Type CDATA #REQUIRED
>
<!ELEMENT W2A_CreateConference (#PCDATA)>
<!ATTLIST W2A_CreateConference
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ConfKey CDATA #REQUIRED
   MaxNumAttendee CDATA #REQUIRED
   Selection CDATA #IMPLIED
   TollFreeNum CDATA #IMPLIED
   TollNum CDATA #IMPLIED
   AccessCode CDATA #IMPLIED
   SubscriberCode CDATA #IMPLIED
   ParticipantPass CDATA #IMPLIED
   ExtUserID CDATA #IMPLIED
   AnonymousID CDATA #IMPLIED
   NBRCallerID CDATA #IMPLIED
>
<!ELEMENT A2W_RspCreateConference (#PCDATA)>
<!ATTLIST A2W_RspCreateConference
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
   PSTNMerge CDATA #IMPLIED
>
<!ELEMENT W2A_CloseConference (#PCDATA)>
<!ATTLIST W2A_CloseConference
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   KeepAlive CDATA #IMPLIED
>
<!ELEMENT A2W_RspCloseConference (#PCDATA)>
<!ATTLIST A2W_RspCloseConference
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_Callout (PhoneNum)>
<!ATTLIST W2A_Callout
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtCallID CDATA #REQUIRED
```

```
   Privilege CDATA #REQUIRED
   MaxWaitingTime CDATA #REQUIRED
   ParticipantType CDATA #IMPLIED
   JoinMode CDATA #IMPLIED
>
<!ELEMENT A2W_RspCallout (#PCDATA)>
<!ATTLIST A2W_RspCallout
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_DropCall (#PCDATA)>
<!ATTLIST W2A_DropCall
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtCallID CDATA #REQUIRED
>
<!ELEMENT A2W_RspDropCall (#PCDATA)>
<!ATTLIST A2W_RspDropCall
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_ChangeCallPrivilege (#PCDATA)>
<!ATTLIST W2A_ChangeCallPrivilege
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtCallID CDATA #REQUIRED
   Privilege CDATA #REQUIRED
>
<!ELEMENT A2W_RspChangeCallPrivilege (#PCDATA)>
<!ATTLIST A2W_RspChangeCallPrivilege
   MsgID CDATA #REQUIRED
   Privilege CDATA #IMPLIED
   Result CDATA #REQUIRED
>
<!ELEMENT A2W_NotifySpeakingStatus (#PCDATA)>
<!ATTLIST A2W_NotifySpeakingStatus
   MsgID CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtCallIDs CDATA #REQUIRED
   SpeakingFlags CDATA #REQUIRED
>
<!ELEMENT W2A_GetConfInfo (#PCDATA)>
```

```
<!ATTLIST W2A_GetConfInfo
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
>
<!ELEMENT A2W_RspGetConfInfo (#PCDATA)>
<!ATTLIST A2W_RspGetConfInfo
   MsgID CDATA #REQUIRED
   ConfStatus CDATA #REQUIRED
   NumAttendee CDATA #REQUIRED
   NumCallIn CDATA #REQUIRED
   NumCallBack CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_GetCallInfo (#PCDATA)>
<!ATTLIST W2A_GetCallInfo
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtCallID CDATA #REQUIRED
>
<!ELEMENT A2W_RspGetCallInfo (#PCDATA)>
<!ATTLIST A2W_RspGetCallInfo
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
   Status CDATA #REQUIRED
   Type CDATA #REQUIRED
>
<!ELEMENT W2A_UpdateConference (#PCDATA)>
<!ATTLIST W2A_UpdateConference
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ActionCode CDATA #REQUIRED
   ExtCallID CDATA #IMPLIED
>
<!ELEMENT A2W_RspUpdateConference (#PCDATA)>
<!ATTLIST A2W_RspUpdateConference
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_RspMessageError (#PCDATA)>
<!ATTLIST W2A_RspMessageError
   MsgID CDATA #REQUIRED
```

```
   MsgName CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_AuthenticateAccount (#PCDATA)>
<!ATTLIST W2A_AuthenticateAccount
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   TollFreeNum CDATA #IMPLIED
   TollNum CDATA #REQUIRED
   AccessCode CDATA #IMPLIED
   SubscriberCode CDATA #REQUIRED
   ParticipantPass CDATA #REQUIRED
   ExtUserID CDATA #IMPLIED
>
<!ELEMENT A2W_RspAuthenticateAccount (#PCDATA)>
<!ATTLIST A2W_RspAuthenticateAccount
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_CreateAccount (#PCDATA)>
<!ATTLIST W2A_CreateAccount
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   TollFreeNum CDATA #IMPLIED
   TollNum CDATA #REQUIRED
   AccessCode CDATA #IMPLIED
   SubscriberCode CDATA #REQUIRED
   ParticipantPass CDATA #REQUIRED
   MaxNumAttendee CDATA #REQUIRED
   RollCall CDATA #REQUIRED
   WbxHostID CDATA #IMPLIED
   WbxSiteID CDATA #IMPLIED
   ExtUserID CDATA #IMPLIED
>
<!ELEMENT A2W_RspCreateAccount (#PCDATA)>
<!ATTLIST A2W_RspCreateAccount
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_UpdateAccount (#PCDATA)>
<!ATTLIST W2A_UpdateAccount
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   TollFreeNum_old CDATA #IMPLIED
```

```
   TollNum_old CDATA #REQUIRED
   AccessCode_old CDATA #IMPLIED
   SubscriberCode_old CDATA #REQUIRED
   ParticipantPass_old CDATA #REQUIRED
   TollFreeNum_new CDATA #IMPLIED
   TollNum_new CDATA #REQUIRED
   AccessCode_new CDATA #IMPLIED
   SubscriberCode_new CDATA #REQUIRED
   ParticipantPass_new CDATA #REQUIRED
   MaxNumAttendee_new CDATA #REQUIRED
   RollCall_new CDATA #REQUIRED
>
<!ELEMENT A2W_RspUpdateAccount (#PCDATA)>
<!ATTLIST A2W_RspUpdateAccount
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_DeleteAccount (#PCDATA)>
<!ATTLIST W2A_DeleteAccount
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   TollFreeNum CDATA #IMPLIED
   TollNum CDATA #IMPLIED
   AccessCode CDATA #IMPLIED
   SubscriberCode CDATA #IMPLIED
   ParticipantPass CDATA #IMPLIED
   WbxSiteID CDATA #IMPLIED
>
<!ELEMENT A2W_RspDeleteAccount (#PCDATA)>
<!ATTLIST A2W_RspDeleteAccount
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT A2W_NotifyUserEnter (#PCDATA)>
<!ATTLIST A2W_NotifyUserEnter
   MsgID CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtCallID CDATA #REQUIRED
   AttendeeID CDATA #IMPLIED
   ParticipantType CDATA #IMPLIED
   DID CDATA #IMPLIED
   ActionCode CDATA #IMPLIED
   ActionResult CDATA #IMPLIED
>
```

```
<!ELEMENT W2A_RspNotifyUserEnter (#PCDATA)>
<!ATTLIST W2A_RspNotifyUserEnter
   MsgID CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtCallID CDATA #REQUIRED
   Result CDATA #REQUIRED
   AttendeeID CDATA #IMPLIED
>
<!ELEMENT A2W_NotifyUserChange (#PCDATA)>
<!ATTLIST A2W_NotifyUserChange
   MsgID CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtCallID CDATA #REQUIRED
   Status CDATA #REQUIRED
   ParticipantType CDATA #IMPLIED
   DID CDATA #IMPLIED
   ActionCode CDATA #IMPLIED
   ActionResult CDATA #IMPLIED
   AttendeeID   CDATA #IMPLIED
>
<!ELEMENT A2W_NotifyConferenceChange (#PCDATA)>
<!ATTLIST A2W_NotifyConferenceChange
   MsgID CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   Status CDATA #REQUIRED
>
<!ELEMENT A2W_RspMessageError (#PCDATA)>
<!ATTLIST A2W_RspMessageError
   MsgID CDATA #REQUIRED
   MsgName CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT PhoneNum (CountryCode?, AreaCode, LocalNumber, Extension?)>
<!ELEMENT CountryCode (#PCDATA)>
<!ELEMENT AreaCode (#PCDATA)>
<!ELEMENT LocalNumber (#PCDATA)>
<!ELEMENT Extension (#PCDATA)>

<!ELEMENT W2A_CreateSubConference (#PCDATA)>
<!ATTLIST W2A_CreateSubConference
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtSubConfID CDATA #REQUIRED
```

```
>
<!ELEMENT A2W_RspCreateSubConference (#PCDATA)>
<!ATTLIST A2W_RspCreateSubConference
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT A2W_NotifySubConferenceChange (#PCDATA)>
<!ATTLIST A2W_NotifySubConferenceChange
   MsgID CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtSubConfID CDATA #REQUIRED
   Status CDATA #REQUIRED
>
<!ELEMENT W2A_CloseSubConference (#PCDATA)>
<!ATTLIST W2A_CloseSubConference
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtSubConfID CDATA #REQUIRED
>
<!ELEMENT A2W_RspCloseSubConference (#PCDATA)>
<!ATTLIST A2W_RspCloseSubConference
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_AddToSubConference (#PCDATA)>
<!ATTLIST W2A_AddToSubConference
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtSubConfID CDATA #REQUIRED
   ExtCallID   CDATA #REQUIRED
   ModeratorFlag CDATA #REQUIRED
>
<!ELEMENT A2W_RspAddToSubConference (#PCDATA)>
<!ATTLIST A2W_RspAddToSubConference
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>

<!ELEMENT W2A_RemoveFromSubConference (#PCDATA)>
<!ATTLIST W2A_RemoveFromSubConference
   MsgID CDATA #REQUIRED
   WbxHostName CDATA #REQUIRED
```

```
   ExtConfID CDATA #REQUIRED
   ExtSubConfID CDATA #REQUIRED
   ExtCallID   CDATA #REQUIRED
>
<!ELEMENT A2W_RspRemoveFromSubConference (#PCDATA)>
<!ATTLIST A2W_RspRemoveFromSubConference
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT A2W_NotifyJoinSubConference (#PCDATA)>
<!ATTLIST A2W_NotifyJoinSubConference
   MsgID CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtSubConfID CDATA #REQUIRED
   ExtCallID CDATA #REQUIRED
   ModeratorFlag CDATA #REQUIRED
>
<!ELEMENT A2W_NotifySubConfCallChange (#PCDATA)>
<!ATTLIST A2W_NotifySubConfCallChange
   MsgID CDATA #REQUIRED
   ExtConfID CDATA #REQUIRED
   ExtSubConfID CDATA #REQUIRED
   ExtCallID CDATA #REQUIRED
   ActionFlag CDATA #REQUIRED
   Status CDATA #REQUIRED
>
<!ELEMENT A2W_CreateConnection (#PCDATA)>
<!ATTLIST A2W_CreateConnection
   MsgID CDATA #REQUIRED
   webExID CDATA #REQUIRED
   SiteName CDATA #REQUIRED
   samlResponse CDATA #REQUIRED
   AdaURL CDATA #REQUIRED
>
<!ELEMENT W2A_RspCreateConnection (#PCDATA)>
<!ATTLIST W2A_RspCreateConnection
   MsgID CDATA #REQUIRED
   Result CDATA #REQUIRED
>
<!ELEMENT W2A_RspNotifyUserChange (#PCDATA)>
<!ATTLIST W2A_RspNotifyUserChange
   MsgID CDATA #REQUIRED
   ExtCallID CDATA #REQUIRED
   Result CDATA #REQUIRED
```

```
   ExtConfID CDATA #REQUIRED
   AttendeeID CDATA #IMPLIED
>
```

B.3.2 XSD

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
   <xs:element name="A2W_NotifyConferenceChange">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
               <xs:attribute name="Status" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_CreateConnection">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="SiteName" type="xs:string"
use="required"/>
               <xs:attribute name="AdaURL" type="xs:string"
use="required"/>
               <xs:attribute name="webExID" type="xs:string"
use="required"/>
               <xs:attribute name="samlResponse" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_NotifyJoinSubConference">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="ExtSubConfID" type="xs:string"
use="required"/>
```

```
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="ExtCallID" type="xs:string"
use="required"/>
            <xs:attribute name="ModeratorFlag" type="xs:string"
use="required"/>
            <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="A2W_NotifySpeakingStatus">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="SpeakingFlags" type="xs:string"
use="required"/>
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="ExtCallIDs" type="xs:string"
use="required"/>
            <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="A2W_NotifySubConfCallChange">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="Status" type="xs:string"
use="required"/>
            <xs:attribute name="ExtSubConfID" type="xs:string"
use="required"/>
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="ActionFlag" type="xs:string"
use="required"/>
            <xs:attribute name="ExtCallID" type="xs:string"
use="required"/>
            <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
```

```
   <xs:element name="A2W_NotifySubConferenceChange">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="Status" type="xs:string"
use="required"/>
               <xs:attribute name="ExtSubConfID" type="xs:string"
use="required"/>
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_NotifyUserChange">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="AttendeeID" type="xs:string"/>
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
               <xs:attribute name="ExtCallID" type="xs:string"
use="required"/>
               <xs:attribute name="Status" type="xs:string"
use="required"/>
               <xs:attribute name="ParticipantType" type="xs:string"/>
               <xs:attribute name="DID" type="xs:string"/>
               <xs:attribute name="ActionCode" type="xs:string"/>
               <xs:attribute name="ActionResult" type="xs:string"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_NotifyUserEnter">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
               <xs:attribute name="ExtCallID" type="xs:string"
use="required"/>
```

```
                <xs:attribute name="AttendeeID" type="xs:string"/>
                <xs:attribute name="ParticipantType" type="xs:string"/>
                <xs:attribute name="DID" type="xs:string"/>
                <xs:attribute name="ActionCode" type="xs:string"/>
                <xs:attribute name="ActionResult" type="xs:string"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
<xs:element name="A2W_RspAddToSubConference">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute name="MsgID" type="xs:string"
use="required"/>
                <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
<xs:element name="A2W_RspAuthenticateAccount">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute name="MsgID" type="xs:string"
use="required"/>
                <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
<xs:element name="A2W_RspCallout">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute name="MsgID" type="xs:string"
use="required"/>
                <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
<xs:element name="A2W_RspChangeCallPrivilege">
```

```
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="Privilege" type="xs:string"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_RspCloseConference">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_RspCloseSubConference">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_RspCreateAccount">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
```

```
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_RspCreateConference">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
               <xs:attribute name="PSTNMerge" type="xs:string"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_RspCreateSubConference">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_RspDeleteAccount">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_RspDropCall">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
```

```
            <xs:attribute name="Result" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="A2W_RspGetBridgeInfo">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="Result" type="xs:string"
use="required"/>
            <xs:attribute name="Type" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="A2W_RspGetBridgeStatus">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="Result" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="A2W_RspGetCallInfo">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="Result" type="xs:string"
use="required"/>
            <xs:attribute name="Status" type="xs:string"
use="required"/>
            <xs:attribute name="Type" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
```

```
   </xs:element>
   <xs:element name="A2W_RspGetConfInfo">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="ConfStatus" type="xs:string"
use="required"/>
               <xs:attribute name="NumAttendee" type="xs:string"
use="required"/>
               <xs:attribute name="NumCallIn" type="xs:string"
use="required"/>
               <xs:attribute name="NumCallBack" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_RspGetResource">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="ResourceType" type="xs:string"
use="required"/>
               <xs:attribute name="Resource" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="A2W_RspMessageError">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="MsgName" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
```

```
            </xs:simpleContent>
         </xs:complexType>
      </xs:element>
      <xs:element name="A2W_RspRemoveFromSubConference">
         <xs:complexType>
            <xs:simpleContent>
               <xs:extension base="xs:string">
                  <xs:attribute name="MsgID" type="xs:string"
use="required"/>
                  <xs:attribute name="Result" type="xs:string"
use="required"/>
               </xs:extension>
            </xs:simpleContent>
         </xs:complexType>
      </xs:element>
      <xs:element name="A2W_RspReset">
         <xs:complexType>
            <xs:simpleContent>
               <xs:extension base="xs:string">
                  <xs:attribute name="MsgID" type="xs:string"
use="required"/>
                  <xs:attribute name="Result" type="xs:string"
use="required"/>
               </xs:extension>
            </xs:simpleContent>
         </xs:complexType>
      </xs:element>
      <xs:element name="A2W_RspUpdateAccount">
         <xs:complexType>
            <xs:simpleContent>
               <xs:extension base="xs:string">
                  <xs:attribute name="MsgID" type="xs:string"
use="required"/>
                  <xs:attribute name="Result" type="xs:string"
use="required"/>
               </xs:extension>
            </xs:simpleContent>
         </xs:complexType>
      </xs:element>
      <xs:element name="A2W_RspUpdateConference">
         <xs:complexType>
            <xs:simpleContent>
               <xs:extension base="xs:string">
                  <xs:attribute name="MsgID" type="xs:string"
use="required"/>
```

```
            <xs:attribute name="Result" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="AreaCode" type="xs:string"/>
<xs:element name="Confirmation">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="Result" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="CountryCode" type="xs:string"/>
<xs:element name="Extension" type="xs:string"/>
<xs:element name="LocalNumber" type="xs:string"/>
<xs:element name="PhoneNum">
   <xs:complexType>
      <xs:sequence>
         <xs:element ref="CountryCode" minOccurs="0"/>
         <xs:element ref="AreaCode" minOccurs="0"/>
         <xs:element ref="LocalNumber"/>
         <xs:element ref="Extension" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
</xs:element>
<xs:element name="TransID" type="xs:string"/>
<xs:element name="W2A_AddToSubConference">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="ExtSubConfID" type="xs:string"
use="required"/>
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
            <xs:attribute name="ExtCallID" type="xs:string"
use="required"/>
            <xs:attribute name="ModeratorFlag" type="xs:string"
use="required"/>
```

```
            <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="W2A_AuthenticateAccount">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
            <xs:attribute name="TollFreeNum" type="xs:string"/>
            <xs:attribute name="TollNum" type="xs:string"/>
            <xs:attribute name="AccessCode" type="xs:string"/>
            <xs:attribute name="SubscriberCode" type="xs:string"
use="required"/>
            <xs:attribute name="ParticipantPass" type="xs:string"
use="required"/>
            <xs:attribute name="ExtUserID" type="xs:string"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="W2A_Callout">
   <xs:complexType>
      <xs:sequence>
         <xs:element ref="PhoneNum"/>
      </xs:sequence>
      <xs:attribute name="MsgID" type="xs:string" use="required"/>
      <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
      <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
      <xs:attribute name="ExtCallID" type="xs:string"
use="required"/>
      <xs:attribute name="Privilege" type="xs:string"
use="required"/>
      <xs:attribute name="MaxWaitingTime" type="xs:string"
use="required"/>
      <xs:attribute name="ParticipantType" type="xs:string"/>
      <xs:attribute name="JoinMode" type="xs:string"/>
   </xs:complexType>
</xs:element>
<xs:element name="W2A_ChangeCallPrivilege">
```

```
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
               <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
               <xs:attribute name="ExtCallID" type="xs:string"
use="required"/>
               <xs:attribute name="Privilege" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="W2A_CloseConference">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
               <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
               <xs:attribute name="KeepAlive" type="xs:string"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="W2A_CloseSubConference">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="ExtSubConfID" type="xs:string"
use="required"/>
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
               <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
```

```
   <xs:element name="W2A_CreateAccount">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
               <xs:attribute name="TollFreeNum" type="xs:string"/>
               <xs:attribute name="TollNum" type="xs:string"/>
               <xs:attribute name="AccessCode" type="xs:string"/>
               <xs:attribute name="SubscriberCode" type="xs:string"
use="required"/>
               <xs:attribute name="ParticipantPass" type="xs:string"
use="required"/>
               <xs:attribute name="MaxNumAttendee" type="xs:string"
use="required"/>
               <xs:attribute name="RollCall" type="xs:string"
use="required"/>
               <xs:attribute name="WbxHostID" type="xs:string"/>
               <xs:attribute name="WbxSiteID" type="xs:string"/>
               <xs:attribute name="ExtUserID" type="xs:string"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="W2A_CreateConference">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MaxNumAttendee" type="xs:string"
use="required"/>
               <xs:attribute name="SubscriberCode" type="xs:string"/>
               <xs:attribute name="ConfKey" type="xs:string"
use="required"/>
               <xs:attribute name="TollFreeNum" type="xs:string"/>
               <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
               <xs:attribute name="AccessCode" type="xs:string"/>
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="ExtUserID" type="xs:string"/>
               <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
               <xs:attribute name="NBRCallerID" type="xs:string"/>
               <xs:attribute name="ParticipantPass" type="xs:string"/>
               <xs:attribute name="TollNum" type="xs:string"/>
```

```
            <xs:attribute name="AnonymousID" type="xs:string"/>
            <xs:attribute name="Selection" type="xs:string"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="W2A_CreateSubConference">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="ExtSubConfID" type="xs:string"
use="required"/>
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
            <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="W2A_DeleteAccount">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
            <xs:attribute name="TollFreeNum" type="xs:string"/>
            <xs:attribute name="TollNum" type="xs:string"/>
            <xs:attribute name="AccessCode" type="xs:string"/>
            <xs:attribute name="SubscriberCode" type="xs:string"/>
            <xs:attribute name="ParticipantPass" type="xs:string"/>
            <xs:attribute name="WbxSiteID" type="xs:string"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="W2A_DropCall">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
```

```
            <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
            <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
            <xs:attribute name="ExtCallID" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="W2A_GetBridgeInfo">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="W2A_GetBridgeStatus">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
         </xs:extension>
      </xs:simpleContent>
   </xs:complexType>
</xs:element>
<xs:element name="W2A_GetCallInfo">
   <xs:complexType>
      <xs:simpleContent>
         <xs:extension base="xs:string">
            <xs:attribute name="MsgID" type="xs:string"
use="required"/>
            <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
            <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
            <xs:attribute name="ExtCallID" type="xs:string"
use="required"/>
         </xs:extension>
```

```
        </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="W2A_GetConfInfo">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
               <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="W2A_GetResource">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
               <xs:attribute name="ResourceType" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="W2A_RemoveFromSubConference">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="ExtSubConfID" type="xs:string"
use="required"/>
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
               <xs:attribute name="ExtCallID" type="xs:string"
use="required"/>
               <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
```

```
      </xs:complexType>
   </xs:element>
   <xs:element name="W2A_Reset">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="W2A_RspMessageError">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="MsgName" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="W2A_RspNotifyUserEnter">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
               <xs:attribute name="ExtCallID" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
               <xs:attribute name="AttendeeID" type="xs:string"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="W2A_UpdateAccount">
      <xs:complexType>
```

```
        <xs:simpleContent>
           <xs:extension base="xs:string">
              <xs:attribute name="MsgID" type="xs:string"
use="required"/>
              <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
              <xs:attribute name="TollFreeNum_old" type="xs:string"/>
              <xs:attribute name="TollNum_old" type="xs:string"/>
              <xs:attribute name="AccessCode_old" type="xs:string"/>
              <xs:attribute name="SubscriberCode_old" type="xs:string"
use="required"/>
              <xs:attribute name="ParticipantPass_old" type="xs:string"
use="required"/>
              <xs:attribute name="TollFreeNum_new" type="xs:string"/>
              <xs:attribute name="TollNum_new" type="xs:string"/>
              <xs:attribute name="AccessCode_new" type="xs:string"/>
              <xs:attribute name="SubscriberCode_new" type="xs:string"
use="required"/>
              <xs:attribute name="ParticipantPass_new" type="xs:string"
use="required"/>
              <xs:attribute name="MaxNumAttendee_new" type="xs:string"
use="required"/>
              <xs:attribute name="RollCall_new" type="xs:string"
use="required"/>
           </xs:extension>
        </xs:simpleContent>
     </xs:complexType>
  </xs:element>
  <xs:element name="W2A_UpdateConference">
     <xs:complexType>
        <xs:simpleContent>
           <xs:extension base="xs:string">
              <xs:attribute name="MsgID" type="xs:string"
use="required"/>
              <xs:attribute name="WbxHostName" type="xs:string"
use="required"/>
              <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
              <xs:attribute name="ActionCode" type="xs:string"
use="required"/>
              <xs:attribute name="ExtCallID" type="xs:string"/>
           </xs:extension>
        </xs:simpleContent>
     </xs:complexType>
  </xs:element>
  <xs:element name="W2A_RspCreateConnection">
     <xs:complexType>
        <xs:simpleContent>
```

```
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="W2A_RspNotifyUserChange">
      <xs:complexType>
         <xs:simpleContent>
            <xs:extension base="xs:string">
               <xs:attribute name="MsgID" type="xs:string"
use="required"/>
               <xs:attribute name="ExtCallID" type="xs:string"
use="required"/>
               <xs:attribute name="Result" type="xs:string"
use="required"/>
               <xs:attribute name="ExtConfID" type="xs:string"
use="required"/>
               <xs:attribute name="AttendeeID" type="xs:string"/>
            </xs:extension>
         </xs:simpleContent>
      </xs:complexType>
   </xs:element>
   <xs:element name="WbxTSPSchema">
      <xs:complexType>
         <xs:sequence>
            <xs:element ref="TransID"/>
            <xs:choice minOccurs="0">
               <xs:element ref="Confirmation"/>
            </xs:choice>
            <xs:choice minOccurs="0" maxOccurs="unbounded">
               <xs:element ref="W2A_Reset"/>
               <xs:element ref="A2W_RspReset"/>
               <xs:element ref="W2A_GetBridgeStatus"/>
               <xs:element ref="A2W_RspGetBridgeStatus"/>
               <xs:element ref="W2A_GetResource"/>
               <xs:element ref="A2W_RspGetResource"/>
               <xs:element ref="W2A_GetBridgeInfo"/>
               <xs:element ref="A2W_RspGetBridgeInfo"/>
               <xs:element ref="W2A_CreateConference"/>
               <xs:element ref="A2W_RspCreateConference"/>
               <xs:element ref="W2A_CloseConference"/>
               <xs:element ref="A2W_RspCloseConference"/>
```

```
            <xs:element ref="W2A_Callout"/>
            <xs:element ref="A2W_RspCallout"/>
            <xs:element ref="W2A_DropCall"/>
            <xs:element ref="A2W_RspDropCall"/>
            <xs:element ref="W2A_ChangeCallPrivilege"/>
            <xs:element ref="A2W_RspChangeCallPrivilege"/>
            <xs:element ref="W2A_GetConfInfo"/>
            <xs:element ref="A2W_RspGetConfInfo"/>
            <xs:element ref="W2A_GetCallInfo"/>
            <xs:element ref="A2W_RspGetCallInfo"/>
            <xs:element ref="W2A_UpdateConference"/>
            <xs:element ref="A2W_RspUpdateConference"/>
            <xs:element ref="W2A_RspMessageError"/>
            <xs:element ref="W2A_AuthenticateAccount"/>
            <xs:element ref="A2W_RspAuthenticateAccount"/>
            <xs:element ref="W2A_CreateAccount"/>
            <xs:element ref="A2W_RspCreateAccount"/>
            <xs:element ref="W2A_UpdateAccount"/>
            <xs:element ref="A2W_RspUpdateAccount"/>
            <xs:element ref="W2A_DeleteAccount"/>
            <xs:element ref="A2W_RspDeleteAccount"/>
            <xs:element ref="A2W_NotifyUserEnter"/>
            <xs:element ref="W2A_RspNotifyUserEnter"/>
            <xs:element ref="A2W_NotifyUserChange"/>
            <xs:element ref="A2W_NotifyConferenceChange"/>
            <xs:element ref="A2W_RspMessageError"/>
            <xs:element ref="A2W_NotifySpeakingStatus"/>
            <xs:element ref="W2A_CreateSubConference"/>
            <xs:element ref="A2W_RspCreateSubConference"/>
            <xs:element ref="W2A_CloseSubConference"/>
            <xs:element ref="A2W_RspCloseSubConference"/>
            <xs:element ref="W2A_AddToSubConference"/>
            <xs:element ref="A2W_RspAddToSubConference"/>
            <xs:element ref="W2A_RemoveFromSubConference"/>
            <xs:element ref="A2W_RspRemoveFromSubConference"/>
            <xs:element ref="A2W_NotifySubConferenceChange"/>
            <xs:element ref="A2W_NotifyJoinSubConference"/>
            <xs:element ref="A2W_NotifySubConfCallChange"/>
            <xs:element ref="A2W_CreateConnection"/>
            <xs:element ref="W2A_RspCreateConnection"/>
            <xs:element ref="W2A_RspNotifyUserChange"/>
        </xs:choice>
    </xs:sequence>
    <xs:attribute name="Name" type="xs:string" use="required"/>
```

```
            <xs:attribute name="Version" type="xs:string" use="required"/>
         </xs:complexType>
      </xs:element>
   </xs:schema>
```

Glossary

A

Adapter

TSP includes a plug-in telephony adapter architecture allowing plug-in adapters to be developed for new telephony servers without the need to customize underlying code.

Adapter application

An adapter required to run between the TSP partner Web server and the bridge to convert the generic API calls to the bridge's own API calls.

Active Speaker

The participant speaking in a session, identified by a blinking icon.

B

Balancer

See Load balancer.

Bridge

A self-contained server that supports telephony conferencing, i.e., all the software is contained inside and is provided by the vendor.

Bridge Call ID

All calls have an ID assigned by the bridge.

C

Channel

A communication channel, i.e., WebEx servers and the bridge. A channel contains the adapter itself.

Computer name

A unique name assigned to a computer (hardware) on the network.

CSV

A table of Comma Separated Values. Comma separated text data format.

D

Dedicated adapter

An adapter that works with a single CACS server.

DHCP

Dynamic Host Configuration Protocol. IP addresses and related information are dynamically configured. TCP/IP networks are easily configured using DHCP. Duplicate addresses are eliminated, and IP addresses are used effectively by centrally managing their allocation.

DNS

Domain Name Service. A system that creates associations between host names and network addresses on a network.

Domain Name Service

See DNS.

E

External ID

See Message ID.

External Call ID

The ID that the TSP server gives each call.

F

Fail-over feature

The WebEx TSP architecture supports redundant telephony servers to allow meetings in progress to be moved to a backup server.

Firewall

A security gateway on the Internet. On the Internet, internal organization networks are constantly affected by unauthorized access from outside. In order to prevent this, a firewall is placed between external and internal networks to filter all incoming packets according to the provisions of the organization's security policy.

Protects private network security with functions to relay only specific protocol from the Internet to the private network.

Full system operation

An operational command that applies to an entire system.

H

HTTP

HyperText Transfer Protocol. One transmission method that is used to pass through a firewall, this text transfer protocol is used on a Web server. A communication protocol used for transactions between Web servers and Meeting Managers on the Internet.

I

IP Address

A unique 32-bit network address allocated to every node on the Internet.

ISP

Internet Service Provider. A company that provides connection service to the Internet. Also called an "Internet provider" or simply "provider."

L

Load balancer

A device that evenly distributes requests to TSP adapters based on specified criteria.

M

Meeting Center

WebEx HTTP protocol meeting teleconferencing flagship product.

Meeting Center TSP Edition

Meeting Center product for teleconferencing service providers.

Message ID

For each message passed between the TSP server and the adapter, it is given an ID.

N

NBR

Network Based Recording.

P

Permanent connection

A connection method type where communication paths are permanently established in order to facilitate event notifications to a host system. The other connection types are connect-when required and Remote Access Service (RAS).

R

Redundancy configuration

WebEx TSP architecture is designed to support different system fail-over and redundancy or duplicated configurations.

S

Server configuration file

The adapter's host name and HTTP port number redefined in the WebEx server configuration file.

Server redundancy

WebEx TSP architecture is designed to support different system fail-over and redundancy or duplicated configurations.

Silent Join

A feature that allows a Sales Manager to invisibly monitor a session without participating, or informing participants.

SSL

Secure Socket Layer. A protocol standard advocated by Netscape Communications that provides secure communication procedures for applications communicating in TCP/IP.

Sub-conference

A separate session that is created outside a main session by participants of the main session.

System

A general term for a client or server.

T

Task

A single operational unit.

TCP/IP

Transmission Control Protocol/Internet Protocol. A world-standard LAN-based network protocol developed mainly by the United States Department of Defense.

Transaction ID

A sequence number.

TSP

A teleconferencing service provider.

TSP Server

A hardware box that runs the WebEx TSP telephony API.

TSP Server Failover

A state in which the TSP Server does not respond to call-in's and call-out's of a session and which prompts WebEx to switch the ongoing session to a redundant TSP server.

U

UI

User interface.

V

VPN

Virtual Private Network. A virtual private network that seethe Internet. A network that prevents eavesdropping by building virtual private communication lines between communicating parties on the Internet and sending communication data after encrypting it. A normal firewall is set up and security is protected on the Internet through protocol tunneling and encryption.

W

WAN

Wide Area Network. A communication network that connects geographically distant locations.

WBS

WebEx Business Suite - This acronym refers to the version of the WebEx Site.

Index

A

B

C

D

E

F

H

L

M

N

P

R

S

T

U

V

W

X