

[MS-PWVPDP]: PowerPoint Web Viewer Presentation Data Protocol Specification

Intellectual Property Rights Notice for Open Specifications Documentation

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

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

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

Revision Summary

Date	Revision History	Revision Class	Comments
07/13/2009	0.1	Major	Initial Availability
08/28/2009	0.2	Editorial	Revised and edited the technical content
11/06/2009	0.3	Editorial	Revised and edited the technical content
02/19/2010	1.0	Major	Updated and revised the technical content
03/31/2010	1.01	Editorial	Revised and edited the technical content
04/30/2010	1.02	Editorial	Revised and edited the technical content
06/07/2010	1.03	Editorial	Revised and edited the technical content
06/29/2010	1.04	Minor	Clarified the meaning of the technical content.
07/23/2010	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
09/27/2010	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	1.05	Major	Significantly changed the technical content.
12/17/2010	1.05	No change	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1 Introduction	8
1.1 Glossary	8
1.2 References	9
1.2.1 Normative References	9
1.2.2 Informative References	10
1.3 Protocol Overview (Synopsis)	10
1.4 Relationship to Other Protocols	10
1.5 Prerequisites/Preconditions	11
1.6 Applicability Statement	11
1.7 Versioning and Capability Negotiation	11
1.8 Vendor-Extensible Fields	11
1.9 Standards Assignments	11
2 Messages	12
2.1 Transport	12
2.2 Common Message Syntax	12
2.2.1 Namespaces	12
2.2.2 Messages	12
2.2.3 Elements	12
2.2.4 Complex Types	13
2.2.4.1 ArrayOfanyType	15
2.2.4.2 ArrayOfBroadcastAnimationStepData	16
2.2.4.3 ArrayOfCapabilityData	16
2.2.4.4 ArrayOfCT_Anim	16
2.2.4.5 ArrayOfCT_AnimList	17
2.2.4.6 ArrayOfCT_Error	17
2.2.4.7 ArrayOfCT_ImageRegionUse	17
2.2.4.8 ArrayOfCT_ResourceEntry	17
2.2.4.9 ArrayOfCT_Shape	18
2.2.4.10 ArrayOfCT_SlideEntry	18
2.2.4.11 ArrayOfstring	18
2.2.4.12 ArrayOfunsignedInt	19
2.2.4.13 ArrayOfVersion	19
2.2.4.14 BroadcastAnimationStepData	19
2.2.4.15 BroadcastData	20
2.2.4.16 BroadcastFile	20
2.2.4.17 BroadcastHostInfo	21
2.2.4.18 BroadcastProtocolInfo	21
2.2.4.19 CapabilityData	21
2.2.4.20 ClientInfo	22
2.2.4.21 CT_Action	22
2.2.4.22 CT_Anim	22
2.2.4.23 CT_AnimIteration	24
2.2.4.24 CT_AnimList	24
2.2.4.25 CT_AnimRepeatOption	24
2.2.4.26 CT_AnimTiming	25
2.2.4.27 CT_BackgroundDesc	25
2.2.4.28 CT_CustomShow	26
2.2.4.29 CT_EmptyType	26
2.2.4.30 CT_Error	26

2.2.4.31	CT_ErrorInfo	27
2.2.4.32	CT_GenericImageRegion	27
2.2.4.33	CT_Hyperlink	27
2.2.4.34	CT_ImageBackground	28
2.2.4.35	CT_ImageGraph	28
2.2.4.36	CT_ImageRegionUse	29
2.2.4.37	CT_IndexRange	29
2.2.4.38	CT_InteractiveAnimUse	30
2.2.4.39	CT_LayoutPosition	30
2.2.4.40	CT_MotionPath	30
2.2.4.41	CT_MotionPathCurveCmd	31
2.2.4.42	CT_MotionPathLineCmd	31
2.2.4.43	CT_MotionPathMoveCmd	32
2.2.4.44	CT_PresentationShowInfo	32
2.2.4.45	CT_Rectangle	33
2.2.4.46	CT_ResourceEntry	33
2.2.4.47	CT_ScaleBy	34
2.2.4.48	CT_Shape	34
2.2.4.49	CT_ShowProperties	35
2.2.4.50	CT_Slide	35
2.2.4.51	CT_SlideEntry	36
2.2.4.52	CT_SlideLayoutInfo	37
2.2.4.53	CT_SlideShowInfo	37
2.2.4.54	CT_SlideShowTimings	38
2.2.4.55	CT_SolidFill	38
2.2.4.56	CT_Transition	38
2.2.4.57	PptViewingService.BinaryConversionResult	39
2.2.4.58	ServerInfo	40
2.2.4.59	ServiceError	40
2.2.4.60	ServiceResult	41
2.2.4.61	SlideInfoResult	41
2.2.4.62	Version	41
2.2.5	Simple Types	42
2.2.5.1	char	42
2.2.5.2	ClientActions	43
2.2.5.3	duration	43
2.2.5.4	guid	44
2.2.5.5	PowerpointView	44
2.2.5.6	ServerCapability	45
2.2.5.7	ServiceErrorType	45
2.2.5.8	SlideShowState	46
2.2.5.9	ST_AnimRepeatType	46
2.2.5.10	ST_AnimStart	47
2.2.5.11	ST_EffectType	47
2.2.5.12	ST_HideAfterAnimType	48
2.2.6	Attributes	48
2.2.7	Groups	48
2.2.8	Attribute Groups	49
2.3	Directory Service Schema Elements	49
3	Protocol Details	50
3.1	Server Details	50
3.1.1	Abstract Data Model	51

3.1.2	Timers	52
3.1.3	Initialization	52
3.1.4	Message Processing Events and Sequencing Rules	52
3.1.4.1	BinaryConvert	52
3.1.4.1.1	Messages	53
3.1.4.1.1.1	IPptViewingService_BinaryConvert_InputMessage	53
3.1.4.1.1.2	IPptViewingService_BinaryConvert_OutputMessage	53
3.1.4.1.2	Elements	53
3.1.4.1.2.1	BinaryConvert	53
3.1.4.1.2.2	BinaryConvertResponse	54
3.1.4.1.3	Complex Types	54
3.1.4.1.4	Simple Types	54
3.1.4.1.5	Attributes	54
3.1.4.1.6	Groups	54
3.1.4.1.7	Attribute Groups	54
3.1.4.2	BroadcastGetData	54
3.1.4.2.1	Messages	55
3.1.4.2.1.1	IPptViewingService_BroadcastGetData_InputMessage	55
3.1.4.2.1.2	IPptViewingService_BroadcastGetData_OutputMessage	55
3.1.4.2.2	Elements	55
3.1.4.2.2.1	BroadcastGetData	55
3.1.4.2.2.2	BroadcastGetDataResponse	55
3.1.4.2.3	Complex Types	56
3.1.4.2.3.1	BroadcastUser	56
3.1.4.2.4	Simple Types	56
3.1.4.2.5	Attributes	56
3.1.4.2.6	Groups	56
3.1.4.2.7	Attribute Groups	56
3.1.4.3	BroadcastJoinSession	56
3.1.4.3.1	Messages	57
3.1.4.3.1.1	IPptViewingService_BroadcastJoinSession_InputMessage	57
3.1.4.3.1.2	IPptViewingService_BroadcastJoinSession_OutputMessage	57
3.1.4.3.2	Elements	57
3.1.4.3.2.1	BroadcastJoinSession	57
3.1.4.3.2.2	BroadcastJoinSessionResponse	58
3.1.4.3.3	Complex Types	58
3.1.4.3.4	Simple Types	58
3.1.4.3.5	Attributes	58
3.1.4.3.6	Groups	58
3.1.4.3.7	Attribute Groups	58
3.1.4.4	BroadcastUnjoinSession	58
3.1.4.4.1	Messages	59
3.1.4.4.1.1	IPptViewingService_BroadcastUnjoinSession_InputMessage	59
3.1.4.4.1.2	IPptViewingService_BroadcastUnjoinSession_OutputMessage	59
3.1.4.4.2	Elements	59
3.1.4.4.2.1	BroadcastUnjoinSession	59
3.1.4.4.2.2	BroadcastUnjoinSessionResponse	59
3.1.4.4.3	Complex Types	60
3.1.4.4.4	Simple Types	60
3.1.4.4.5	Attributes	60
3.1.4.4.6	Groups	60
3.1.4.4.7	Attribute Groups	60
3.1.4.5	GetNotesHtmlServiceResult	60

3.1.4.5.1	Messages	61
3.1.4.5.1.1	IPptViewingService_GetNotesHtmlServiceResult_InputMessage	61
3.1.4.5.1.2	IPptViewingService_GetNotesHtmlServiceResult_OutputMessage	61
3.1.4.5.2	Elements	61
3.1.4.5.2.1	GetNotesHtmlServiceResult	61
3.1.4.5.2.2	GetNotesHtmlServiceResultResponse	62
3.1.4.5.3	Complex Types	62
3.1.4.5.4	Simple Types	62
3.1.4.5.5	Attributes	62
3.1.4.5.6	Groups	62
3.1.4.5.7	Attribute Groups	62
3.1.4.6	GetPresentation	62
3.1.4.6.1	Messages	63
3.1.4.6.1.1	IPptViewingService_GetPresentation_InputMessage	63
3.1.4.6.1.2	IPptViewingService_GetPresentation_OutputMessage	63
3.1.4.6.2	Elements	63
3.1.4.6.2.1	GetPresentation	63
3.1.4.6.2.2	GetPresentationResponse	63
3.1.4.6.3	Complex Types	64
3.1.4.6.4	Simple Types	64
3.1.4.6.5	Attributes	64
3.1.4.6.6	Groups	64
3.1.4.6.7	Attribute Groups	64
3.1.4.7	GetPresentationInfoXml	64
3.1.4.7.1	Messages	64
3.1.4.7.1.1	IPptViewingService_GetPresentationInfoXml_InputMessage	65
3.1.4.7.1.2	IPptViewingService_GetPresentationInfoXml_OutputMessage	65
3.1.4.7.2	Elements	65
3.1.4.7.2.1	GetPresentationInfoXml	65
3.1.4.7.2.2	GetPresentationInfoXmlResponse	65
3.1.4.7.3	Complex Types	66
3.1.4.7.4	Simple Types	66
3.1.4.7.5	Attributes	66
3.1.4.7.6	Groups	66
3.1.4.7.7	Attribute Groups	66
3.1.4.8	GetSlide	66
3.1.4.8.1	Messages	66
3.1.4.8.1.1	IPptViewingService_GetSlide_InputMessage	66
3.1.4.8.1.2	IPptViewingService_GetSlide_OutputMessage	67
3.1.4.8.2	Elements	67
3.1.4.8.2.1	GetSlide	67
3.1.4.8.2.2	GetSlideResponse	67
3.1.4.8.3	Complex Types	68
3.1.4.8.4	Simple Types	68
3.1.4.8.5	Attributes	68
3.1.4.8.6	Groups	68
3.1.4.8.7	Attribute Groups	68
3.1.4.9	GetSlideInfo	68
3.1.4.9.1	Messages	68
3.1.4.9.1.1	IPptViewingService_GetSlideInfo_InputMessage	68
3.1.4.9.1.2	IPptViewingService_GetSlideInfo_OutputMessage	69
3.1.4.9.2	Elements	69
3.1.4.9.2.1	GetSlideInfo	69

3.1.4.9.2.2	GetSlideInfoResponse	69
3.1.4.9.3	Complex Types	69
3.1.4.9.4	Simple Types	69
3.1.4.9.5	Attributes	70
3.1.4.9.6	Groups	70
3.1.4.9.7	Attribute Groups	70
3.1.5	Timer Events	70
3.1.6	Other Local Events	70
3.2	Client Details	70
3.2.1	Abstract Data Model	70
3.2.2	Timers	70
3.2.3	Initialization	70
3.2.4	Message Processing Events and Sequencing Rules	70
3.2.5	Timer Events	70
3.2.6	Other Local Events	70
4	Protocol Examples	71
4.1	Interaction Between Protocol Client and Protocol Server	71
4.1.1	Interaction Between Attendee Protocol Client and Protocol Server	74
5	Security	77
5.1	Security Considerations for Implementers	77
5.2	Index of Security Parameters	77
6	Appendix A: Full WSDL	78
7	Appendix B: Product Behavior	96
8	Change Tracking	97
9	Index	98

1 Introduction

This document specifies the PowerPoint Web Viewer Presentation Data Protocol, which enables a protocol client to obtain information about **presentation** content on a protocol server.

1.1 Glossary

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

broadcast
Hypertext Transfer Protocol (HTTP)
Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)

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

cross-site scripting
hyperlink
main master slide
motion path
presentation
presentation broadcast
presentation slide
shape
slide layout
slide show
slide show broadcast
SOAP (Simple Object Access Protocol)
SOAP action
SOAP body
SOAP fault
SOAP message
ToolTip
URL (Uniform Resource Locator)
view
Web site
WSDL (Web Services Description Language)
WSDL message
WSDL operation
XML namespace
XML Schema

The following terms are specific to this document:

hit test region: An area that is configured to execute an action, such as opening the destination of a hyperlink, in response to user input, such as a mouse click or key press.

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

1.2 References

1.2.1 Normative References

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

[HTML] World Wide Web Consortium, "HTML 4.01 Specification", December 1999, <http://www.w3.org/TR/html4/>

[MS-OMPWHP] Microsoft Corporation, "[Office Mobile PowerPoint Web Handler Protocol Specification](#)", July 2009.

[MS-PWBHPS] Microsoft Corporation, "[PowerPoint Web Broadcast Host Protocol Specification](#)", July 2009.

[MS-PWEDPS] Microsoft Corporation, "[PowerPoint Web Editor Data Protocol Specification](#)", July 2009.

[MS-PWVRSC] Microsoft Corporation, "[PowerPoint Web Viewer Rendered Static Content Structure Specification](#)", July 2009.

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

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

[RFC4627] Crockford, D., "The application/json Media Type for Javascript Object Notation (JSON)", RFC 4627, July 2006, <http://www.ietf.org/rfc/rfc4627.txt>

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

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

[WSA1.0] World Wide Web Consortium, "Web Services Addressing 1.0 - WSDL Binding", 29 May 2006, <http://www.w3.org/TR/ws-addr-wsdl/>

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

[XMLNS] World Wide Web Consortium, "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation 8 December 2009, <http://www.w3.org/TR/REC-xml-names/>

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

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

1.2.2 Informative References

[ISO/IEC-29500:2008] International Organization for Standardization, "Information technology -- Document description and processing languages -- Office Open XML File Formats -- Parts 1-4", Publicly Available Standards, <http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html>

[MS-GLOS] Microsoft Corporation, "[Windows Protocols Master Glossary](#)", March 2007.

[MS-OFCGLOS] Microsoft Corporation, "[Microsoft Office Master Glossary](#)", June 2008.

[MS-PPT] Microsoft Corporation, "[PowerPoint Binary File Format \(.ppt\) Structure Specification](#)", June 2008.

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

[RFC2822] Resnick, P., Ed., "Internet Message Format", STD 11, RFC 2822, April 2001, <http://www.ietf.org/rfc/rfc2822.txt>

[SOAP1.2/2] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H.F., "SOAP Version 1.2 Part 2: Adjuncts", W3C Recommendation, June 2003, <http://www.w3.org/TR/2003/REC-soap12-part2-20030624>

1.3 Protocol Overview (Synopsis)

This protocol enables a protocol client to send a request to retrieve information about presentation content from the protocol server.

To facilitate this, the protocol allows the protocol client to request specific pieces of content from a presentation stored on the protocol server. In a **presentation slide** contained within a presentation, a protocol client can retrieve information describing the presentation slide contents as well as images of the presentation slide contents.

The protocol also enables an attendee client to send requests to a protocol server allowing the client to join a **broadcast** session, and to retrieve data about the state of a broadcast on the protocol server.

1.4 Relationship to Other Protocols

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

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

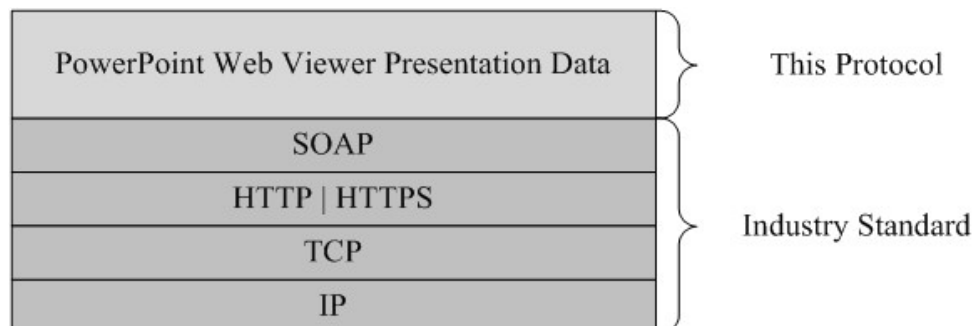


Figure 1: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against a **Web site(1)** that is identified by a **URL** that is known by protocol clients. The protocol server endpoint is formed by appending "/_vti_bin/ppt/view.svc" to the URL of the Web site, for example http://www.contoso.com/Repository/_vti_bin/ppt/view.svc.

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

1.6 Applicability Statement

This protocol is designed to retrieve information about presentations that are stored on the protocol server.

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

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

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

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

2.1 Transport

Protocol servers **MUST** support SOAP over HTTP. Protocol servers **SHOULD** additionally support SOAP over HTTPS for securing communication with clients. Protocol servers **SHOULD** support JSON [\[RFC4627\]](#) over HTTP and HTTPS.

Protocol messages **MUST** be formatted as specified either in [\[SOAP1.1\]](#) section 4 or in [\[SOAP1.2/1\]](#) section 5. Protocol server faults **MUST** be returned either using HTTP Status Codes, as specified in [\[RFC2616\]](#) section 10 or using **SOAP faults**, as specified in either [\[SOAP1.1\]](#) section 4.4 or in [\[SOAP1.2/1\]](#) section 5.4.

2.2 Common Message Syntax

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

2.2.1 Namespaces

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

Prefix	Namespace URI	Reference
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
tns	http://tempuri.org/	
tns1	http://schemas.datacontract.org/2004/07/p	
wsaw	http://www.w3.org/2006/05/addressing/wsdl	[WSA1.0]
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1] [XMLSCHEMA2]

2.2.2 Messages

None.

2.2.3 Elements

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

2.2.4 Complex Types

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

Complex Type	Description
ArrayOfanyType	A complex type that specifies a list of <code>xs:anyType</code> [XMLSCHEMA1] section 3.4.7 elements.
ArrayOfBroadcastAnimationStepData	A complex type that specifies a list of BroadcastAnimationStepData elements.
ArrayOfCapabilityData	Reserved. MUST be ignored.
ArrayOfCT_Anim	A complex type that specifies a list of CT_Anim elements.
ArrayOfCT_AnimList	A complex type that specifies a list of CT_AnimList elements.
ArrayOfCT_Error	A complex type that specifies a list of CT_Error elements.
ArrayOfCT_ImageRegionUse	A complex type that specifies a list of CT_ImageRegionUse elements.
ArrayOfCT_ResourceEntry	A complex type that specifies a list of CT_ResourceEntry elements.
ArrayOfCT_Shape	A complex type that specifies a list of CT_Shape elements.
ArrayOfCT_SlideEntry	A complex type that specifies a list of CT_SlideEntry elements.
ArrayOfstring	A complex type that specifies a list of <code>xs:string</code> [XMLSCHEMA2] section 3.2.1 elements.
ArrayOfunsignedInt	A complex type that specifies a list of <code>xs:unsignedInt</code> [XMLSCHEMA2] section 3.3.22 elements.
ArrayOfVersion	Reserved. MUST be ignored.
BroadcastAnimationStepData	A complex type that specifies a step in an animation timeline.
BroadcastData	A complex type that specifies the slide show broadcast state data.
BroadcastFile	Reserved. MUST be ignored.
BroadcastHostInfo	Reserved. MUST be ignored.
BroadcastProtocolInfo	Reserved. MUST be ignored.
CapabilityData	Reserved. MUST be ignored.
ClientInfo	Reserved. MUST be ignored.
CT_Action	A complex type that specifies information about a slide navigation action.
CT_Anim	A complex type that specifies a shape animation.

Complex Type	Description
CT_AnimIteration	A complex type that specifies timing information for the individual regions of an animation.
CT_AnimList	A complex type that specifies a list of animations.
CT_AnimRepeatOption	A complex type that specifies the animation repeat settings for an animation.
CT_AnimTiming	A complex type that specifies timing information for an animation.
CT_BackgroundDesc	A complex type that specifies the background appearance of a presentation slide.
CT_CustomShow	A complex type that specifies a custom slide show in the form of a sequence of slide identifiers.
CT_EmptyType	A complex type that represents an empty type. Can be used where no specific element needs to be present.
CT_Error	Reserved. MUST be ignored.
CT_ErrorInfo	Reserved. MUST be ignored.
CT_GenericImageRegion	A complex type that specifies the image region for a single image relative to a larger image cluster.
CT_Hyperlink	A complex type that specifies a hyperlink .
CT_ImageBackground	A complex type that specifies the image region for a background image relative to a larger image cluster.
CT_ImageGraph	A complex type that specifies information about images that are required to display an animation, consisting of the primary image and an optional list of image variations.
CT_ImageRegionUse	A complex type that specifies an image region along with its layout position.
CT_IndexRange	A complex type that specifies a zero-based index range of a list.
CT_InteractiveAnimUse	A complex type that specifies information about an animation trigger.
CT_LayoutPosition	A complex type that specifies a two dimensional position.
CT_MotionPath	A complex type that specifies a motion path .
CT_MotionPathCurveCmd	A complex type that specifies a curved segment of a motion path. The curve is described as a Bezier curve with three control points.
CT_MotionPathLineCmd	A complex type that specifies a linear segment of a motion path.
CT_MotionPathMoveCmd	A complex type that specifies the next point of a motion path.

Complex Type	Description
CT_PresentationShowInfo	A complex type that specifies the detailed information for showing a presentation.
CT_Rectangle	A complex type that specifies a rectangle.
CT_ResourceEntry	A complex type that specifies a resource description.
CT_ScaleBy	A complex type that specifies the scale factors for animations that use scale effects.
CT_Shape	A complex type that specifies a shape object.
CT_ShowProperties	A complex type that specifies the slides that will be shown in a slide show .
CT_Slide	A complex type that specifies a presentation slide.
CT_SlideEntry	A complex type that specifies the basic information of a presentation slide.
CT_SlideLayoutInfo	A complex type that specifies layout information for a presentation slide.
CT_SlideShowInfo	A complex type that specifies detailed information for showing a presentation slide.
CT_SlideShowTimings	A complex type that specifies timing information for a slide show.
CT_SolidFill	A complex type that specifies the information for a solid fill.
CT_Transition	A complex type that specifies a slide transition.
PptViewingService.BinaryConversionResult	A complex type that specifies the result of a binary conversion.
ServerInfo	Reserved. MUST be ignored.
ServiceError	A complex type that specifies error information returned by the protocol server to a protocol client.
ServiceResult	A complex type that specifies the result of a protocol method. The protocol server returns this type to the protocol client containing either a successful Result element or an Error element.
SlideInfoResult	Reserved. MUST be ignored.
Version	Reserved. MUST be ignored.

2.2.4.1 ArrayOfanyType

Namespace: <http://schemas.microsoft.com/2003/10/Serialization/Arrays>

A complex type that specifies a list of xs:anyType [\[XMLSCHEMA1\]](#) section 3.4.7 elements.

```
<xs:complexType name="ArrayOfanyType">
  <xs:sequence>
```

```
<xs:element minOccurs="0" maxOccurs="unbounded" name="anyType" nillable="true"
type="xs:anyType"/>
</xs:sequence>
</xs:complexType>
```

anyType: Each element MUST specify a `xs:anyType` [\[XMLSCHEMA1\]](#) section 3.4.7.

2.2.4.2 ArrayOfBroadcastAnimationStepData

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data>

A complex type that specifies a list of [BroadcastAnimationStepData](#) elements.

```
<xs:complexType name="ArrayOfBroadcastAnimationStepData">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="BroadcastAnimationStepData"
nillable="true" type="tns:BroadcastAnimationStepData"/>
  </xs:sequence>
</xs:complexType>
```

BroadcastAnimationStepData: A complex type that specifies a step in an animation timeline.

2.2.4.3 ArrayOfCapabilityData

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data>

Reserved. MUST be ignored.

```
<xs:complexType name="ArrayOfCapabilityData">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="CapabilityData" nillable="true"
type="tns:CapabilityData"/>
  </xs:sequence>
</xs:complexType>
```

CapabilityData: Reserved. MUST be ignored.

2.2.4.4 ArrayOfCT_Anim

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies a list of [CT_Anim](#) elements.

```
<xs:complexType name="ArrayOfCT_Anim">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_Anim" nillable="true"
type="tns:CT_Anim"/>
  </xs:sequence>
</xs:complexType>
```


CT_Anim: Each element MUST specify a CT_Anim.

2.2.4.5 ArrayOfCT_AnimList

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies a list of [CT_AnimList](#) elements.

```
<xs:complexType name="ArrayOfCT_AnimList">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_AnimList" nillable="true"
type="tns:CT_AnimList"/>
  </xs:sequence>
</xs:complexType>
```

CT_AnimList: Each element MUST specify a CT_AnimList.

2.2.4.6 ArrayOfCT_Error

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies a list of [CT_Error](#) elements.

```
<xs:complexType name="ArrayOfCT_Error">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_Error" nillable="true"
type="tns:CT_Error"/>
  </xs:sequence>
</xs:complexType>
```

CT_Error: Each element MUST specify a CT_Error.

2.2.4.7 ArrayOfCT_ImageRegionUse

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies a list of [CT_ImageRegionUse](#) elements.

```
<xs:complexType name="ArrayOfCT_ImageRegionUse">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_ImageRegionUse" nillable="true"
type="tns:CT_ImageRegionUse"/>
  </xs:sequence>
</xs:complexType>
```

CT_ImageRegionUse: Each element MUST specify a CT_ImageRegionUse.

2.2.4.8 ArrayOfCT_ResourceEntry

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies a list of [CT_ResourceEntry](#) elements.

```
<xs:complexType name="ArrayOfCT_ResourceEntry">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_ResourceEntry" nillable="true"
type="tns:CT_ResourceEntry"/>
  </xs:sequence>
</xs:complexType>
```

CT_ResourceEntry: Each element MUST specify a CT_ResourceEntry.

2.2.4.9 ArrayOfCT_Shape

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies a list of [CT_Shape](#) elements.

```
<xs:complexType name="ArrayOfCT_Shape">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_Shape" nillable="true"
type="tns:CT_Shape"/>
  </xs:sequence>
</xs:complexType>
```

CT_Shape: Each element MUST specify a CT_Shape.

2.2.4.10 ArrayOfCT_SlideEntry

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies a list of [CT_SlideEntry](#) elements.

```
<xs:complexType name="ArrayOfCT_SlideEntry">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_SlideEntry" nillable="true"
type="tns:CT_SlideEntry"/>
  </xs:sequence>
</xs:complexType>
```

CT_SlideEntry: Each element MUST specify a CT_SlideEntry.

2.2.4.11 ArrayOfstring

Namespace: <http://schemas.microsoft.com/2003/10/Serialization/Arrays>

A complex type that specifies a list of xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 elements.

```
<xs:complexType name="ArrayOfstring">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
type="xs:string"/>
  </xs:sequence>
```

```
</xs:complexType>
```

string: Each element MUST specify a xs:string [\[XMLSCHEMA2\]](#) section 3.2.1.

2.2.4.12 ArrayOfunsignedInt

Namespace: <http://schemas.microsoft.com/2003/10/Serialization/Arrays>

A complex type that specifies a list of xs:unsignedInt [\[XMLSCHEMA2\]](#) section 3.3.22 elements.

```
<xs:complexType name="ArrayOfunsignedInt">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="unsignedInt"
type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
```

unsignedInt: Each element MUST specify an xs:unsignedInt [\[XMLSCHEMA2\]](#) section 3.3.22.

2.2.4.13 ArrayOfVersion

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data>

Reserved. MUST be ignored.

```
<xs:complexType name="ArrayOfVersion">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="Version" nillable="true"
type="tns:Version"/>
  </xs:sequence>
</xs:complexType>
```

Version: Reserved. MUST be ignored.

2.2.4.14 BroadcastAnimationStepData

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data>

A complex type that specifies a step in an animation timeline.

```
<xs:complexType name="BroadcastAnimationStepData">
  <xs:sequence>
    <xs:element minOccurs="0" name="Step" type="xs:int"/>
    <xs:element minOccurs="0" name="TimelineId" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

Step: An xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the step number in the given animation timeline. MUST be present.

TimelineId: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies an identifier of the animation timeline. MUST be present.

2.2.4.15 BroadcastData

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data

A complex type that specifies the slide show broadcast state data.

```
<xs:complexType name="BroadcastData">
  <xs:sequence>
    <xs:element minOccurs="0" name="AnimationStepDataList" nillable="true"
type="tns:ArrayOfBroadcastAnimationStepData"/>
    <xs:element minOccurs="0" name="HostToken" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="SequenceNumber" type="xs:int"/>
    <xs:element minOccurs="0" name="SlideId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="SlideShowState" type="tns:SlideShowState"/>
  </xs:sequence>
</xs:complexType>
```

AnimationStepDataList: An [ArrayOfBroadcastAnimationStepData](#) element that specifies the current step in each of the animation timelines.

HostToken: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the token returned by PowerPoint Web Broadcast Host protocol [\[MS-PWBHPS\]](#) server corresponding to the presentation that is uploaded by the protocol client. This element MUST be present.

SequenceNumber: An xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies a monotonically increasing sequence number. This element MUST be present.

SlideId: An xs:unsignedInt [\[XMLSCHEMA2\]](#) section 3.3.22 element that specifies the identifier of the presentation slide. This element MUST be present.

SlideShowState: A [SlideShowState](#) element that specifies the current state of the slide show. This element MUST be present.

2.2.4.16 BroadcastFile

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data

Reserved. MUST be ignored.

```
<xs:complexType name="BroadcastFile">
  <xs:sequence>
    <xs:element minOccurs="0" name="FilePath" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="PresentationId" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

FilePath: Reserved. MUST be ignored.

PresentationId: Reserved. MUST be ignored.

2.2.4.17 BroadcastHostInfo

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data

Reserved. MUST be ignored.

```
<xs:complexType name="BroadcastHostInfo">
  <xs:sequence>
    <xs:element minOccurs="0" name="HostUrl" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="ProtocolVersion" nillable="true" type="tns:Version"/>
  </xs:sequence>
</xs:complexType>
```

HostUrl: Reserved. MUST be ignored.

ProtocolVersion: Reserved. MUST be ignored.

2.2.4.18 BroadcastProtocolInfo

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data

Reserved. MUST be ignored.

```
<xs:complexType name="BroadcastProtocolInfo">
  <xs:sequence>
    <xs:element minOccurs="0" name="ClientInfo" nillable="true" type="tns:ClientInfo"/>
    <xs:element minOccurs="0" name="VersionList" nillable="true" type="tns:ArrayOfVersion"/>
  </xs:sequence>
</xs:complexType>
```

ClientInfo: Reserved. MUST be ignored.

VersionList: Reserved. MUST be ignored.

2.2.4.19 CapabilityData

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data

Reserved. MUST be ignored.

```
<xs:complexType name="CapabilityData">
  <xs:sequence>
    <xs:element minOccurs="0" name="Capability" type="tns:ServerCapability"/>
    <xs:element minOccurs="0" name="Value" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

Capability: Reserved. MUST be ignored.

Value: Reserved. MUST be ignored.

2.2.4.20 ClientInfo

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data

Reserved. MUST be ignored.

```
<xs:complexType name="ClientInfo">
  <xs:sequence>
    <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Version" nillable="true" type="tns:Version"/>
  </xs:sequence>
</xs:complexType>
```

Name: Reserved. MUST be ignored.

Version: Reserved. MUST be ignored.

2.2.4.21 CT_Action

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies information about a slide navigation action.

```
<xs:complexType name="CT_Action">
  <xs:sequence>
    <xs:element name="rct" nillable="true" type="tns:CT_Rectangle"/>
    <xs:element name="sldId" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
```

rct: A [CT_Rectangle](#) element that specifies the **hit test region** that triggers this action.

sldId: An xs:unsignedInt [\[XMLSCHEMA2\]](#) section 3.3.22 element that specifies the identifier of the target presentation slide that is associated with this action. MUST have a maximum value of less than 2147483648 and it MUST have a minimum value of greater than or equal to 256.

2.2.4.22 CT_Anim

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies a shape animation.

```
<xs:complexType name="CT_Anim">
  <xs:sequence>
    <xs:element name="autoRev" type="xs:boolean"/>
    <xs:element name="dir" type="xs:int"/>
    <xs:element name="effectId" type="xs:unsignedInt"/>
    <xs:element name="effectType" type="tns:ST_EffectType"/>
    <xs:element name="empImgs" nillable="true" type="tns:CT_ImageGraph"/>
    <xs:element name="hideAfter" type="tns:ST_HideAfterAnimType"/>
    <xs:element name="iter" nillable="true" type="tns:CT_AnimIteration"/>
    <xs:element name="mp" nillable="true" type="tns:CT_MotionPath"/>
  </xs:sequence>
</xs:complexType>
```

```

<xs:element name="repeat" nillable="true" type="tns:CT_AnimRepeatOption"/>
<xs:element name="rewind" type="xs:boolean"/>
<xs:element name="rotAng" type="xs:int"/>
<xs:element name="scaleBy" nillable="true" type="tns:CT_ScaleBy"/>
<xs:element name="sp" type="xs:unsignedInt"/>
<xs:element name="timing" nillable="true" type="tns:CT_AnimTiming"/>
<xs:element name="transp" type="xs:unsignedByte"/>
</xs:sequence>
</xs:complexType>

```

autoRev: An optional xs:boolean [\[XMLSCHEMA2\]](#) section 3.2.2 element that specifies whether this animation automatically plays in reverse after the initial animation.

dir: An optional xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the variant of an animation effect for this animation.

effectId: An xs:unsignedInt [\[XMLSCHEMA2\]](#) section 3.3.22 element that specifies the identifier of an animation effect for this animation.

effectType: An [ST_EffectType](#) element that specifies the type of animation effect for this animation.

empImgs: An optional [CT_ImageGraph](#) element that specifies image information to be used by this animation.

hideAfter: An optional [ST_HideAfterAnimType](#) element that specifies the behavior to apply to the shape once the animation ends.

iter: An optional [CT_AnimIteration](#) element that specifies child element timing information to be used by this animation.

mp: An optional [CT_MotionPath](#) element that specifies the motion path to be used for this animation.

repeat: An optional [CT_AnimRepeatOption](#) element that specifies repeat information to be used by this animation.

rewind: An optional xs:boolean [\[XMLSCHEMA2\]](#) section 3.2.2 element that specifies whether the shape returns to its original position when the animation has finished.

rotAng: An optional xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the rotation amount, in degrees, by which the shape rotates, for animations that use a rotate component.

scaleBy: An optional [CT_ScaleBy](#) element that specifies scale information to be used by this animation.

sp: An xs:unsignedInt [\[XMLSCHEMA2\]](#) section 3.3.22 element that specifies the identifier of the shape to which this animation applies.

timing: A [CT_AnimTiming](#) element that specifies the timing information for this animation.

transp: An optional xs:unsignedByte [\[XMLSCHEMA2\]](#) section 3.3.24 element that specifies the amount of transparency, in percent, to apply to the shape for animations that use a transparency component. MUST be less than or equal to 100.

2.2.4.23 CT_AnimIteration

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies timing information for the individual regions of an animation.

```
<xs:complexType name="CT_AnimIteration">
  <xs:sequence>
    <xs:element name="interval" type="xs:int"/>
    <xs:element name="reversed" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
```

interval: An xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the time, in milliseconds, between each animation region's start time. MUST be greater than or equal to 0.

reversed: An xs:boolean [\[XMLSCHEMA2\]](#) section 3.2.2 element that specifies whether the animation regions play back in reverse order.

2.2.4.24 CT_AnimList

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies a list of animations.

```
<xs:complexType name="CT_AnimList">
  <xs:sequence>
    <xs:element name="anim" nillable="true" type="tns:ArrayOfCT_Anim"/>
    <xs:element name="id" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
```

anim: An [ArrayOfCT_Anim](#) element that specifies a list of animations.

id: An xs:unsignedInt [\[XMLSCHEMA2\]](#) section 3.3.22 element that specifies the unique identifier of the list.

2.2.4.25 CT_AnimRepeatOption

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies the animation repeat settings for an animation.

```
<xs:complexType name="CT_AnimRepeatOption">
  <xs:sequence>
    <xs:element name="count" type="xs:int"/>
    <xs:element name="type" type="tns:ST_AnimRepeatType"/>
  </xs:sequence>
</xs:complexType>
```


count: An optional `xs:int` [XMLSCHEMA2] section 3.3.17 element that specifies the number of repeats for this animation. MUST be greater than or equal to 0. MUST be ignored if the value of the **type** element is **clk** or **sld**.

type: An [ST_AnimRepeatType](#) element that specifies how this animation repeats.

2.2.4.26 CT_AnimTiming

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies timing information for an animation.

```
<xs:complexType name="CT_AnimTiming">
  <xs:sequence>
    <xs:element name="off" type="xs:unsignedInt"/>
    <xs:element name="spd" nillable="true" type="xs:string"/>
    <xs:element name="st" type="tns:ST_AnimStart"/>
  </xs:sequence>
</xs:complexType>
```

off: An optional `xs:unsignedInt` [XMLSCHEMA2] section 3.3.22 element that specifies the time offset, in milliseconds, of the start of the animation. MUST be greater than or equal to zero.

spd: An optional `xs:string` [XMLSCHEMA2] section 3.2.1 element that specifies the duration of an animation effect. It MUST be a value from the following table:

Value	Meaning
def	The animation effect uses the default duration for its type.
clk	The animation effect lasts until the next click.
sld	The animation effect lasts until the end of the presentation slide.
A string of digits that matches the regular expression pattern: [0-9]+	The animation effect duration is the integer value, in milliseconds, represented by the string in base-10 form.

st: An optional [ST_AnimStart](#) element that specifies how the animation start time is determined.

2.2.4.27 CT_BackgroundDesc

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies the background appearance of a presentation slide.

```
<xs:complexType name="CT_BackgroundDesc">
  <xs:sequence>
    <xs:element name="Item" nillable="true" type="xs:anyType"/>
  </xs:sequence>
</xs:complexType>
```

Item: An `xs:anyType` [\[XMLSCHEMA1\]](#) section 3.4.7 element that specifies the background appearance of a presentation slide. The child element MUST be [CT_ImageBackground](#) or [CT_SolidFill](#). A `CT_ImageBackground` child element specifies the image to use for the background. A `CT_SolidFill` child element specifies the color fill to use for the background.

2.2.4.28 CT_CustomShow

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies a custom slide show in the form of a sequence of slide identifiers.

```
<xs:complexType name="CT_CustomShow">
  <xs:sequence>
    <xs:element xmlns:q4="http://schemas.microsoft.com/2003/10/Serialization/Arrays" name="s"
      nillable="true" type="q4:ArrayOfunsignedInt"/>
  </xs:sequence>
</xs:complexType>
```

s: An [ArrayOfunsignedInt](#) element that specifies a list of slide identifiers. Each identifier in the list MUST have a maximum value of less than 2147483648 and MUST have a minimum value of greater than or equal to 256.

2.2.4.29 CT_EmptyType

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that represents an empty type. Can be used where no specific element needs to be present.

```
<xs:complexType name="CT_EmptyType">
  <xs:sequence/>
</xs:complexType>
```

2.2.4.30 CT_Error

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

Reserved. MUST be ignored.

```
<xs:complexType name="CT_Error">
  <xs:sequence>
    <xs:element name="id" type="xs:unsignedInt"/>
    <xs:element xmlns:q1="http://schemas.microsoft.com/2003/10/Serialization/Arrays" name="p"
      nillable="true" type="q1:ArrayOfstring"/>
  </xs:sequence>
</xs:complexType>
```

id: MUST be ignored.

p: MUST be ignored.

2.2.4.31 CT_ErrorInfo

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

Reserved. MUST be ignored.

```
<xs:complexType name="CT_ErrorInfo">
  <xs:sequence>
    <xs:element name="err" nillable="true" type="tns:CT_Error"/>
    <xs:element name="schmVer" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
```

err: MUST be ignored.

schmVer: MUST be ignored.

2.2.4.32 CT_GenericImageRegion

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies the image region for a single image relative to a larger image cluster.

```
<xs:complexType name="CT_GenericImageRegion">
  <xs:sequence>
    <xs:element name="h" type="xs:int"/>
    <xs:element name="imgID" nillable="true" type="xs:string"/>
    <xs:element name="w" type="xs:int"/>
    <xs:element name="x" type="xs:int"/>
    <xs:element name="y" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

h: An xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the height, in pixels, of the image region. MUST be greater than or equal to 1.

imgID: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the identifier for this image region.

w: An xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the width, in pixels, of the image region. MUST be greater than or equal to 1.

x: An xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the specifies the x coordinate, in pixels, of the upper-left corner of the image region. MUST be greater than or equal to 0.

y: An xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the y coordinate, in pixels, of the upper-left corner of the image region. MUST be greater than or equal to 0.

2.2.4.33 CT_Hyperlink

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies a hyperlink.

```
<xs:complexType name="CT_Hyperlink">
  <xs:sequence>
    <xs:element name="rct" nillable="true" type="tns:CT_Rectangle"/>
    <xs:element name="tip" nillable="true" type="xs:string"/>
    <xs:element name="url" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

rct: A [CT_Rectangle](#) element that specifies the hit test region for this hyperlink. MUST be present.

tip: An optional `xs:string` [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the text content for the hyperlink's **ToolTip**.

url: An `xs:string` [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the target URL for the hyperlink.

2.2.4.34 CT_ImageBackground

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies the image region for a background image relative to a larger image cluster.

```
<xs:complexType name="CT_ImageBackground">
  <xs:sequence>
    <xs:element name="h" type="xs:int"/>
    <xs:element name="imgID" nillable="true" type="xs:string"/>
    <xs:element name="w" type="xs:int"/>
    <xs:element name="x" type="xs:int"/>
    <xs:element name="y" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

h: An `xs:int` [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the height, in pixels, of the image region. MUST be greater than or equal to 1.

imgID: An `xs:string` [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the identifier for this image region.

w: An `xs:int` [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the width, in pixels, of the image region. MUST be greater than or equal to 1.

x: An `xs:int` [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the x coordinate, in pixels, of the upper-left corner of the image region. MUST be greater than or equal to 0.

y: An `xs:int` [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the y coordinate, in pixels, of the upper-left corner of the image region. MUST be greater than or equal to 0.

2.2.4.35 CT_ImageGraph

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies information about images that are required to display an animation, consisting of the primary image and an optional list of image variations.

```
<xs:complexType name="CT_ImageGraph">
  <xs:sequence>
    <xs:element name="base" nillable="true" type="tns:CT_ImageRegionUse"/>
    <xs:element name="children" nillable="true" type="tns:ArrayOfCT_ImageRegionUse"/>
  </xs:sequence>
</xs:complexType>
```

base: A [CT_ImageRegionUse](#) element that specifies the image region and position for the primary image in the image graph.

children: An [ArrayOfCT_ImageRegionUse](#) element that specifies a list of images and regions for the individual image variations in the image graph.

2.2.4.36 CT_ImageRegionUse

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies an image region along with its layout position.

```
<xs:complexType name="CT_ImageRegionUse">
  <xs:sequence>
    <xs:element name="pos" nillable="true" type="tns:CT_LayoutPosition"/>
    <xs:element name="reg" nillable="true" type="tns:CT_GenericImageRegion"/>
  </xs:sequence>
</xs:complexType>
```

pos: A [CT_LayoutPosition](#) element that specifies the location for the image.

reg: A [CT_GenericImageRegion](#) element that specifies source information about the individual image relative to a larger image cluster.

2.2.4.37 CT_IndexRange

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies a zero-based index range of a list.

```
<xs:complexType name="CT_IndexRange">
  <xs:sequence>
    <xs:element name="end" type="xs:unsignedInt"/>
    <xs:element name="st" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
```

end: An `xs:unsignedInt` [\[XMLSCHEMA2\]](#) section 3.3.22 element that specifies the non-inclusive end index of the range.

st: An `xs:unsignedInt` [\[XMLSCHEMA2\]](#) section 3.3.22 element that specifies the inclusive start index of the range.

2.2.4.38 CT_InteractiveAnimUse

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies information about an animation trigger.

```
<xs:complexType name="CT_InteractiveAnimUse">
  <xs:sequence>
    <xs:element name="rct" nillable="true" type="tns:CT_Rectangle"/>
    <xs:element name="seqId" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
```

rct: A [CT_Rectangle](#) element that specifies the hit test region that triggers the animation.

seqId: An xs:unsignedInt [\[XMLSCHEMA2\]](#) section 3.3.22 element that specifies the identifier of the animation timeline that is triggered when the hit test region is clicked.

2.2.4.39 CT_LayoutPosition

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies a two dimensional position.

```
<xs:complexType name="CT_LayoutPosition">
  <xs:sequence>
    <xs:element name="x" type="xs:int"/>
    <xs:element name="y" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

x: An xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the x coordinate, in pixels, of the position.

y: An xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the y coordinate, in pixels, of the position.

2.2.4.40 CT_MotionPath

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies a motion path.

```
<xs:complexType name="CT_MotionPath">
  <xs:sequence>
    <xs:element xmlns:q2="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
      name="Items" nillable="true" type="q2:ArrayOfanyType"/>
    <xs:element name="rev" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
```

Items: An [ArrayOfanyType](#) element that specifies a motion path. The child elements MUST be [CT_MotionPathMoveCmd](#), [CT_MotionPathLineCmd](#), or [CT_MotionPathCurveCmd](#). A [CT_MotionPathMoveCmd](#) child element specifies the next point of the motion path. A [CT_MotionPathLineCmd](#) child element specifies a linear segment of the motion path. A [CT_MotionPathCurveCmd](#) child element specifies a curve segment of the motion path.

rev: An optional `xs:boolean` [\[XMLSCHEMA2\]](#) section 3.2.2 attribute that specifies whether the motion path is reversed.

2.2.4.41 CT_MotionPathCurveCmd

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies a curved segment of a motion path. The curve is described as a Bezier curve with three control points.

```
<xs:complexType name="CT_MotionPathCurveCmd">
  <xs:sequence>
    <xs:element name="x1" type="xs:int"/>
    <xs:element name="x2" type="xs:int"/>
    <xs:element name="x3" type="xs:int"/>
    <xs:element name="y1" type="xs:int"/>
    <xs:element name="y2" type="xs:int"/>
    <xs:element name="y3" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

x1: An `xs:int` [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the x coordinate, in pixels, of the first control point for the segment.

x2: An `xs:int` [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the x coordinate, in pixels, of the second control point for the segment.

x3: An `xs:int` [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the x coordinate, in pixels, of the third control point for the segment.

y1: An `xs:int` [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the y coordinate, in pixels, of the first control point for the segment.

y2: An `xs:int` [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the y coordinate, in pixels, of the second control point for the segment.

y3: An `xs:int` [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the y coordinate, in pixels, of the third control point for the segment.

2.2.4.42 CT_MotionPathLineCmd

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies a linear segment of a motion path.

```
<xs:complexType name="CT_MotionPathLineCmd">
  <xs:sequence>
    <xs:element name="x" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

```

    <xs:element name="y" type="xs:int"/>
  </xs:sequence>
</xs:complexType>

```

x: An xs:int [XMLSCHEMA2] section 3.3.17 element that specifies the x coordinate, in pixels, of the segment's endpoint.

y: An xs:int [XMLSCHEMA2] section 3.3.17 element that specifies the y coordinate, in pixels, of the segment's endpoint.

2.2.4.43 CT_MotionPathMoveCmd

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies the next point of a motion path.

```

<xs:complexType name="CT_MotionPathMoveCmd">
  <xs:sequence>
    <xs:element name="x" type="xs:int"/>
    <xs:element name="y" type="xs:int"/>
  </xs:sequence>
</xs:complexType>

```

x: An xs:int [XMLSCHEMA2] section 3.3.17 element that specifies the x coordinate, in pixels, of the point.

y: An xs:int [XMLSCHEMA2] section 3.3.17 element that specifies the y coordinate, in pixels, of the point.

2.2.4.44 CT_PresentationShowInfo

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies the detailed information for showing a presentation.

```

<xs:complexType name="CT_PresentationShowInfo">
  <xs:sequence>
    <xs:element name="h" type="xs:int"/>
    <xs:element name="hasCmt" type="xs:boolean"/>
    <xs:element name="rtl" type="xs:boolean"/>
    <xs:element name="schmVer" type="xs:unsignedInt"/>
    <xs:element name="showPr" nillable="true" type="tns:CT_ShowProperties"/>
    <xs:element name="sldLst" nillable="true" type="tns:ArrayOfCT_SlideEntry"/>
    <xs:element name="w" type="xs:int"/>
  </xs:sequence>
</xs:complexType>

```

h: An xs:int [XMLSCHEMA2] section 3.3.17 element that specifies the height of a presentation in pixels. MUST be greater than or equal to 1.

hasCmt: A xs:boolean [XMLSCHEMA2] section 3.2.2 element that specifies whether a presentation has comments.

rtl: Reserved. MUST be ignored.

schmVer: An `xs:unsignedInt` [XMLSCHEMA2] 3.3.22 element that specifies the schema version of the presentation.

showPr: A `CT_ShowProperties` element that specifies the presentation slides that will be shown.

sldLst: An `ArrayOfCT_SlideEntry` element that specifies a list of slide entries.

w: An `xs:int` [XMLSCHEMA2] 3.3.17 element that specifies the width of a presentation in pixels. MUST be greater than or equal to 1.

2.2.4.45 CT_Rectangle

Namespace:

`http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared`

A complex type that specifies a rectangle.

```
<xs:complexType name="CT_Rectangle">
  <xs:sequence>
    <xs:element name="h" type="xs:int"/>
    <xs:element name="l" type="xs:int"/>
    <xs:element name="t" type="xs:int"/>
    <xs:element name="w" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

h: An `xs:int` [XMLSCHEMA2] section 3.3.17 element that specifies height of the rectangle, in pixels. MUST be greater than or equal to 1.

l: An `xs:int` [XMLSCHEMA2] section 3.3.17 element that specifies the x coordinate of the upper-left corner of the rectangle, in pixels.

t: An `xs:int` [XMLSCHEMA2] section 3.3.17 element that specifies the y coordinate of the upper-left corner of the rectangle, in pixels.

w: An `xs:int` [XMLSCHEMA2] section 3.3.17 element that specifies the width of the rectangle, in pixels. MUST be greater than or equal to 1.

2.2.4.46 CT_ResourceEntry

Namespace:

`http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared`

A complex type that specifies a resource description.

```
<xs:complexType name="CT_ResourceEntry">
  <xs:sequence>
    <xs:element name="id" nillable="true" type="xs:string"/>
    <xs:element name="name" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

id: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the unique identifier of the resource description.

name: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the name of the resource description. The name MUST have a file name extension and the file name extension SHOULD be an image format such as ".png" and ".jpg".

2.2.4.47 CT_ScaleBy

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies the scale factors for animations that use scale effects.

```
<xs:complexType name="CT_ScaleBy">
  <xs:sequence>
    <xs:element name="x" type="xs:int"/>
    <xs:element name="y" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

x: An xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the x scale factor, in percentage of original size, for the animation. MUST be greater than or equal to 0.

y: An xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the y scale factor, in percentage of original size, for the animation. MUST be greater than or equal to 0.

2.2.4.48 CT_Shape

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies a shape object.

```
<xs:complexType name="CT_Shape">
  <xs:sequence>
    <xs:element name="bgSp" type="xs:boolean"/>
    <xs:element name="children" nillable="true" type="tns:ArrayOfCT_ImageRegionUse"/>
    <xs:element xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
name="clkTgts" nillable="true" type="q3:ArrayOfanyType"/>
    <xs:element name="id" type="xs:unsignedInt"/>
    <xs:element name="pos" nillable="true" type="tns:CT_LayoutPosition"/>
    <xs:element name="reg" nillable="true" type="tns:CT_GenericImageRegion"/>
  </xs:sequence>
</xs:complexType>
```

bgSp: Reserved. MUST be ignored.

children: An [ArrayOfCT_ImageRegionUse](#) element that specifies the child objects of the shape.

clkTgts: An optional [ArrayOfanyType](#) element that specifies click behaviors for a shape. The child elements MUST be [CT_Hyperlink](#), [CT_Action](#) or [CT_InteractiveAnimUse](#). A CT_Hyperlink child element specifies an external hyperlink click target. A CT_Action child element specifies a click target that navigates to another presentation slide in the presentation. A CT_InteractiveAnimUse child element specifies an animation timeline trigger.

id: An xs:unsignedInt [\[XMLSCHEMA2\]](#) section 3.3.22 element that specifies the unique identifier of the shape.

pos: A [CT_LayoutPosition](#) element that specifies the position of the shape.

reg: A [CT_GenericImageRegion](#) element that specifies the region of the shape.

2.2.4.49 CT_ShowProperties

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies the slides that will be shown in a slide show.

```
<xs:complexType name="CT_ShowProperties">
  <xs:sequence>
    <xs:element name="Item" nillable="true" type="xs:anyType"/>
  </xs:sequence>
</xs:complexType>
```

Item: An xs:anyType [\[XMLSCHEMA1\]](#) section 3.4.7 element that specifies the slides that will be shown in a slide show. This element MUST be [CT_EmptyType](#), [CT_IndexRange](#), or [CT_CustomShow](#). A CT_EmptyType element specifies that all slides will be shown. A CT_IndexRange specifies the range of the slides that will be shown. The range MUST be contained within the presentation's range. A CT_CustomShow element specifies a custom slide show in the form of a sequence of slides. Only slides in the corresponding custom slide show will be shown in sequence. All slides within the custom slide show MUST be within the presentation.

2.2.4.50 CT_Slide

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies a presentation slide.

```
<xs:complexType name="CT_Slide">
  <xs:sequence>
    <xs:element name="animLst" nillable="true" type="tns:CT_AnimList"/>
    <xs:element name="animSeqs" nillable="true" type="tns:ArrayOfCT_AnimList"/>
    <xs:element name="bg" nillable="true" type="tns:CT_BackgroundDesc"/>
    <xs:element name="bgReg" nillable="true" type="tns:CT_GenericImageRegion"/>
    <xs:element name="hidden" type="xs:boolean"/>
    <xs:element name="id" type="xs:unsignedInt"/>
    <xs:element name="layoutInfo" nillable="true" type="tns:CT_SlideLayoutInfo"/>
    <xs:element name="notesId" nillable="true" type="xs:string"/>
    <xs:element name="resCat" nillable="true" type="tns:ArrayOfCT_ResourceEntry"/>
    <xs:element name="spLst" nillable="true" type="tns:ArrayOfCT_Shape"/>
    <xs:element name="ssTimes" nillable="true" type="tns:CT_SlideShowTimings"/>
    <xs:element name="title" nillable="true" type="xs:string"/>
    <xs:element name="trans" nillable="true" type="tns:CT_Transition"/>
  </xs:sequence>
</xs:complexType>
```

animLst: A [CT_AnimList](#) element that specifies the list of animations of the presentation slide.

animSeqs: An [ArrayOfCT_AnimList](#) element that specifies a series of animation sequences.

bg: A [CT_BackgroundDesc](#) element that specifies the background appearance of the presentation slide.

bgReg: A [CT_GenericImageRegion](#) element that specifies the background region of the presentation slide.

hidden: An optional xs:boolean [\[XMLSCHEMA2\]](#) 3.2.2 element that specifies whether the presentation slide is hidden.

id: An xs:unsignedInt [\[XMLSCHEMA2\]](#) section 3.3.22 element that specifies a unique identifier for the presentation slide. MUST have a maximum value of less than 2147483648 and MUST have a minimum value of greater than or equal to 256.

layoutInfo: A [CT_SlideLayoutInfo](#) element that specifies the layout information for the presentation slide.

notesId: An optional xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies a unique identifier for the notes contents of the presentation slide.

resCat: An [ArrayOfCT_ResourceEntry](#) element that specifies a set of resources that are in the presentation slide.

spLst: An [ArrayOfCT_Shape](#) element that specifies the shapes that are included in the presentation slide.

ssTimes: A [CT_SlideShowTimings](#) element that specifies timing information for a slide show.

title: An optional xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the title of the presentation slide.

trans: A [CT_Transition](#) element that specifies the slide transition.

2.2.4.51 CT_SlideEntry

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies the basic information of a presentation slide.

```
<xs:complexType name="CT_SlideEntry">
  <xs:sequence>
    <xs:element name="id" type="xs:unsignedInt"/>
    <xs:element name="sldInfo" nillable="true" type="xs:string"/>
    <xs:element name="title" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

id: An xs:unsignedInt [\[XMLSCHEMA2\]](#) section 3.3.22 element that specifies the unique identifier of the presentation slide. MUST have a maximum value of less than 2147483648 and MUST have a minimum value of greater than or equal to 256.

sldInfo: An xs:string [\[XMLSCHEMA2\]](#) 3.2.1 element that specifies a name that uniquely refers to detailed information of the presentation slide in the presentation.

title: An optional `xs:string` [\[XMLSCHEMA2\]](#) 3.2.1 element that specifies the title of the presentation slide.

2.2.4.52 CT_SlideLayoutInfo

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies layout information for a presentation slide.

```
<xs:complexType name="CT_SlideLayoutInfo">
  <xs:sequence>
    <xs:element name="fTitleLayout" type="xs:boolean"/>
    <xs:element name="layoutId" type="xs:unsignedInt"/>
    <xs:element name="masterId" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
```

fTitleLayout: An `xs:boolean` [\[XMLSCHEMA2\]](#) section 3.2.2 element that specifies whether this slide uses a layout with one title and one subtitle placeholder.

layoutId: An `xs:unsignedInt` [\[XMLSCHEMA2\]](#) 3.3.22 element that specifies the identifier for the **slide layout**.

masterId: An `xs:unsignedInt` [\[XMLSCHEMA2\]](#) 3.3.22 element that specifies the identifier for the **main master slide**.

2.2.4.53 CT_SlideShowInfo

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared>

A complex type that specifies detailed information for showing a presentation slide.

```
<xs:complexType name="CT_SlideShowInfo">
  <xs:sequence>
    <xs:element name="errLst" nillable="true" type="tns:ArrayOfCT_Error"/>
    <xs:element xmlns:q5="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
name="errMsgLst" nillable="true" type="q5:ArrayOfstring"/>
    <xs:element name="schmVer" type="xs:unsignedInt"/>
    <xs:element name="sld" nillable="true" type="tns:CT_Slide"/>
  </xs:sequence>
</xs:complexType>
```

errLst: Reserved. MUST be ignored.

errMsgLst: Reserved. MUST be ignored.

schmVer: An `xs:unsignedInt` [\[XMLSCHEMA2\]](#) 3.3.22 element that specifies the schema version.

sld: A [CT_Slide](#) element that specifies a presentation slide.

2.2.4.54 CT_SlideShowTimings

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies timing information for a slide show.

```
<xs:complexType name="CT_SlideShowTimings">
  <xs:sequence>
    <xs:element name="autoAdv" type="xs:boolean"/>
    <xs:element name="autoAdvTime" type="xs:int"/>
    <xs:element name="clickTimes" nillable="true" type="xs:string"/>
    <xs:element name="manualAdv" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
```

autoAdv: An optional xs:boolean [\[XMLSCHEMA2\]](#) 3.2.2 element that specifies whether a presentation slide will automatically advance when viewed in a slide show.

autoAdvTime: An optional xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the timing, in milliseconds, at which the slide show automatically advances. MUST be greater than or equal to 0.

clickTimes: Reserved. MUST be ignored.

manualAdv: An optional xs:boolean [\[XMLSCHEMA2\]](#) 3.2.2 element that specifies whether the user can manually advance.

2.2.4.55 CT_SolidFill

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies the information for a solid fill.

```
<xs:complexType name="CT_SolidFill">
  <xs:sequence>
    <xs:element name="clr" nillable="true" type="xs:base64Binary"/>
    <xs:element name="transp" type="xs:unsignedByte"/>
  </xs:sequence>
</xs:complexType>
```

clr: An xs:base64Binary [\[XMLSCHEMA2\]](#) 3.2.16 element that specifies the fill color.

transp: An optional xs:unsignedByte [\[XMLSCHEMA2\]](#) 3.3.24 element that specifies the amount of transparency, in percent, of the fill color.

2.2.4.56 CT_Transition

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A complex type that specifies a slide transition.

```
<xs:complexType name="CT_Transition">
  <xs:sequence>
```

```

    <xs:element name="dir" type="xs:unsignedByte"/>
    <xs:element name="dur" type="xs:int"/>
    <xs:element name="type" type="xs:unsignedByte"/>
  </xs:sequence>
</xs:complexType>

```

dir: An xs:unsignedByte [\[XMLSCHEMA2\]](#) 3.3.24 element that specifies whether a cut transition is used. If the value of the **type** element is not 0x00, this element MUST be ignored.

dur: An xs:int [\[XMLSCHEMA2\]](#) section 3.3.17 element that specifies the duration of the transition in milliseconds.

type: An xs:unsignedByte [\[XMLSCHEMA2\]](#) 3.3.24 element that specifies the type of the transition.

Value	Meaning
0x00	The transition is a cut transition. A cut transition immediately changes to the next slide with no visual effect between slides. If the dir element equals the value 0x01, the cut transition first transitions to a black slide before transitioning to the next presentation slide. If the dir element does not equal the value 0x01, the cut transition transitions directly to the next presentation slide.
0x06	The transition is a fade through black transition which fades to a black slide before fading to the next presentation slide.
All Other Values	The transition is a fade to the next presentation slide.

2.2.4.57 PptViewingService.BinaryConversionResult

Namespace: http://schemas.datacontract.org/2004/07/p

A complex type that specifies the result of a binary conversion.

```

<xs:complexType name="PptViewingService.BinaryConversionResult">
  <xs:sequence>
    <xs:element minOccurs="0" name="EditUrl" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="FExists" type="xs:boolean"/>
    <xs:element minOccurs="0" name="ViewUrl" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

```

EditUrl: An optional xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the URL for editing the output presentation.

FExists: A xs:boolean [\[XMLSCHEMA2\]](#) section 3.2.2 element that specifies whether the output presentation already exists.

ViewUrl: An optional xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the URL for viewing the output presentation.

2.2.4.58 ServerInfo

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data

Reserved. MUST be ignored.

```
<xs:complexType name="ServerInfo">
  <xs:sequence>
    <xs:element minOccurs="0" name="CapabilitiesList" nillable="true"
type="tns:ArrayOfCapabilityData"/>
    <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="PresenterServiceUrl" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Version" nillable="true" type="tns:Version"/>
  </xs:sequence>
</xs:complexType>
```

CapabilitiesList: Reserved. MUST be ignored.

Name: Reserved. MUST be ignored.

PresenterServiceUrl: Reserved. MUST be ignored.

Version: Reserved. MUST be ignored.

2.2.4.59 ServiceError

Namespace: http://schemas.datacontract.org/2004/07/p

A complex type that specifies error information returned by the protocol server to a protocol client.

```
<xs:complexType name="ServiceError">
  <xs:sequence>
    <xs:element minOccurs="0" name="Message" nillable="true" type="xs:string"/>
    <xs:element
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.Int
erface" minOccurs="0" name="RecommendedActions" type="q1:ClientActions"/>
    <xs:element minOccurs="0" name="Title" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Type" type="tns:ServiceErrorType"/>
  </xs:sequence>
</xs:complexType>
```

Message: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the error message description. MUST be present. The string length MUST be greater than zero if the **Type** element has a value of [ApplicationError](#).

RecommendedActions: Reserved. MUST be ignored.

Title: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the error title. MUST be present. The string length MUST be greater than zero if the **Type** element has a value of [ApplicationError](#).

Type: A ServiceErrorType element that specifies the error type. MUST be present.

2.2.4.60 ServiceResult

Namespace: http://schemas.datacontract.org/2004/07/p

A complex type that specifies the result of a protocol method. The protocol server returns this type to the protocol client containing either a successful **Result** element or an **Error** element.

```
<xs:complexType name="ServiceResult">
  <xs:sequence>
    <xs:element minOccurs="0" name="Error" nillable="true" type="tns:ServiceError"/>
    <xs:element minOccurs="0" name="Result" nillable="true" type="xs:anyType"/>
  </xs:sequence>
</xs:complexType>
```

Error: An optional [ServiceError](#) element that specifies an error result of a protocol message response. This element **MUST** be present if the **Result** element is not present. Otherwise, this element **MUST NOT** be present.

Result: An optional xs:anyType [\[XMLSCHEMA1\]](#) section 3.4.7 element that specifies a successful result of a protocol message response. This element **MUST** be present if the **Error** element is not present. Otherwise, this element **MUST NOT** be present.

2.2.4.61 SlideInfoResult

Namespace: http://schemas.datacontract.org/2004/07/p

Reserved. **MUST** be ignored.

```
<xs:complexType name="SlideInfoResult">
  <xs:sequence>
    <xs:element minOccurs="0" name="ResourceInfoList" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Slide" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

ResourceInfoList: Reserved. **MUST** be ignored.

Slide: Reserved. **MUST** be ignored.

2.2.4.62 Version

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data

Reserved. **MUST** be ignored.

```
<xs:complexType name="Version">
  <xs:sequence>
    <xs:element minOccurs="0" name="MajorNumber" type="xs:int"/>
    <xs:element minOccurs="0" name="MinorNumber" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

MajorNumber: Reserved. **MUST** be ignored.

MinorNumber: Reserved. MUST be ignored.

2.2.5 Simple Types

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

Simple Type	Description
char	Reserved. The client and server MUST NOT use it as the type of any element. Client and server MUST ignore it if receiving an element of this type.
ClientActions	Reserved. MUST be ignored.
duration	Reserved. The client and server MUST NOT use it as the type of any element. Client and server MUST ignore it if receiving an element of this type.
guid	Reserved. The client and server MUST NOT use it as the type of any element. Client and server MUST ignore it if receiving an element of this type.
PowerpointView	A simple type that specifies an enumeration of all the views of a presentation. This simple type's contents are a restriction of the xs:string [XMLSCHEMA2] section 3.2.1 data type.
ServerCapability	Reserved. MUST be ignored.
ServiceErrorType	A simple type that specifies an enumeration of a set of protocol errors returned by the protocol server to the protocol client.
SlideShowState	A simple type that specifies an enumeration of all the possible slide show states. This simple type's contents are a restriction of the xs:string [XMLSCHEMA2] section 3.2.1 data type.
ST_AnimRepeatType	A simple type that specifies how an animation effect is repeated for a shape . This simple type's contents are a restriction of the xs:string [XMLSCHEMA2] section 3.2.1 data type.
ST_AnimStart	A simple type that specifies when an animation starts. This simple type's contents are a restriction of the xs:string [XMLSCHEMA2] section 3.2.1 data type.
ST_EffectType	A simple type that specifies the type of animation effect for a shape. This simple type's contents are a restriction of the xs:string [XMLSCHEMA2] section 3.2.1 data type.
ST_HideAfterAnimType	A simple type that specifies if and when an animated shape is hidden after the animation effect ends. This simple type's content are a restriction of the xs:string [XMLSCHEMA2] section 3.2.1 data type.

2.2.5.1 char

Namespace: <http://schemas.microsoft.com/2003/10/Serialization/>

Reserved. The client and server MUST NOT use it as the type of any element. Client and server MUST ignore it if receiving an element of this type.

```
<xs:simpleType name="char">  
  <xs:restriction base="xs:int"/>  
</xs:simpleType>
```

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

2.2.5.2 ClientActions

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.Interface

Reserved. MUST be ignored.

```
<xs:simpleType name="ClientActions">
  <xs:list>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="None"/>
        <xs:enumeration value="Dismiss"/>
        <xs:enumeration value="Close"/>
        <xs:enumeration value="OpenInClient"/>
        <xs:enumeration value="Refresh"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:list>
</xs:simpleType>
```

The following table specifies the allowable values for ClientActions:

Value	Meaning
None	Reserved. MUST be ignored.
Dismiss	Reserved. MUST be ignored.
Close	Reserved. MUST be ignored.
OpenInClient	Reserved. MUST be ignored.
Refresh	Reserved. MUST be ignored.

2.2.5.3 duration

Namespace: http://schemas.microsoft.com/2003/10/Serialization/

Reserved. The client and server MUST NOT use it as the type of any element. Client and server MUST ignore it if receiving an element of this type.

```
<xs:simpleType name="duration">
  <xs:restriction base="xs:duration">
    <xs:pattern value="\-?P(\d*D)?(T(\d*H)?(\d*M)?(\d*(\.\d*)?S)?)?"/>
    <xs:minInclusive value="-P10675199DT2H48M5.4775808S"/>
    <xs:maxInclusive value="P10675199DT2H48M5.4775807S"/>
  </xs:restriction>
</xs:simpleType>
```

2.2.5.4 guid

Namespace: http://schemas.microsoft.com/2003/10/Serialization/

Reserved. The client and server MUST NOT use it as the type of any element. Client and server MUST ignore it if receiving an element of this type.

```
<xs:simpleType name="guid">
  <xs:restriction base="xs:string">
    <xs:pattern value="\{8\}-\{4\}-\{4\}-\{4\}-\{12\}" />
  </xs:restriction>
</xs:simpleType>
```

2.2.5.5 PowerpointView

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A simple type that specifies an enumeration of all the views of a presentation. This simple type's contents are a restriction of the xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 data type.

```
<xs:simpleType name="PowerpointView">
  <xs:restriction base="xs:string">
    <xs:enumeration value="ReadingView" />
    <xs:enumeration value="SlideShowView" />
    <xs:enumeration value="EditView" />
    <xs:enumeration value="StaticView" />
    <xs:enumeration value="OutlineView" />
    <xs:enumeration value="AttendeeView" />
  </xs:restriction>
</xs:simpleType>
```

The following table specifies the allowable values for PowerpointView:

Value	Meaning
ReadingView	A view of the presentation contents that has been optimized for reading. This view includes presentation notes, images of presentation slide content and animation information.
SlideShowView	A view of the presentation contents that has been optimized for presenting. This view includes images of presentation slide content and animation information.
EditView	Reserved. MUST be ignored.
StaticView	A view of the presentation that has been optimized for viewing presentation slide contents without animations.
OutlineView	A view of the presentation that has been optimized for viewing presentation slide contents in a text format.
AttendeeView	A view of the presentation that has been optimized for attending a presentation broadcast .

2.2.5.6 ServerCapability

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data

Reserved. MUST be ignored.

```
<xs:simpleType name="ServerCapability">
  <xs:restriction base="xs:string">
    <xs:enumeration value="SupportAudio"/>
    <xs:enumeration value="SupportVideo"/>
    <xs:enumeration value="SupportNotes"/>
    <xs:enumeration value="SupportHandoutMaster"/>
    <xs:enumeration value="FileSizeLimited"/>
    <xs:enumeration value="UseWebDav"/>
    <xs:enumeration value="SessionTimeout"/>
    <xs:enumeration value="SessionIdleTimeout"/>
  </xs:restriction>
</xs:simpleType>
```

The following table specifies the allowable values for ServerCapability:

Value	Meaning
SupportAudio	Reserved. MUST be ignored.
SupportVideo	Reserved. MUST be ignored.
SupportNotes	Reserved. MUST be ignored.
SupportHandoutMaster	Reserved. MUST be ignored.
FileSizeLimited	Reserved. MUST be ignored.
UseWebDav	Reserved. MUST be ignored.
SessionTimeout	Reserved. MUST be ignored.
SessionIdleTimeout	Reserved. MUST be ignored.

2.2.5.7 ServiceErrorType

Namespace: http://schemas.datacontract.org/2004/07/p

A simple type that specifies an enumeration of a set of protocol errors returned by the protocol server to the protocol client.

```
<xs:simpleType name="ServiceErrorType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="UnknownError"/>
    <xs:enumeration value="ApplicationError"/>
    <xs:enumeration value="Timeout"/>
    <xs:enumeration value="ServiceBusy"/>
    <xs:enumeration value="SessionFull"/>
  </xs:restriction>
</xs:simpleType>
```

The following table specifies the allowable values for ServiceErrorType:

Value	Meaning
UnknownError	The protocol server encountered an unknown error.
ApplicationError	The protocol server encountered an application error.
Timeout	The protocol server timed out.
ServiceBusy	The protocol server is busy.
SessionFull	The broadcast session on the protocol server is full.

2.2.5.8 SlideShowState

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data

A simple type that specifies an enumeration of all the possible slide show states. This simple type's contents are a restriction of the xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 data type.

```
<xs:simpleType name="SlideShowState">
  <xs:restriction base="xs:string">
    <xs:enumeration value="NotStartedYet"/>
    <xs:enumeration value="BlackScreen"/>
    <xs:enumeration value="WhiteScreen"/>
    <xs:enumeration value="Normal"/>
    <xs:enumeration value="BroadcastEnded"/>
    <xs:enumeration value="SlideShowEnded"/>
  </xs:restriction>
</xs:simpleType>
```

The following table specifies the allowable values for SlideShowState:

Value	Meaning
NotStartedYet	slide show is not started yet.
BlackScreen	slide show is displaying a black Screen.
WhiteScreen	slide show is displaying a white Screen.
Normal	slide show is displaying presentation slides.
BroadcastEnded	slide show broadcast has ended.
SlideShowEnded	slide show has ended.

2.2.5.9 ST_AnimRepeatType

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A simple type that specifies how an animation effect is repeated for a shape. This simple type's contents are a restriction of the xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 data type.

```

<xs:simpleType name="ST_AnimRepeatType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="cnt"/>
    <xs:enumeration value="clk"/>
    <xs:enumeration value="sld"/>
  </xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for ST_AnimRepeatType:

Value	Meaning
cnt	The animation effect is repeated a specific number of times.
clk	The animation effect is repeated until the next click.
sld	The animation effect is repeated until the end of the current presentation slide.

2.2.5.10 ST_AnimStart

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A simple type that specifies when an animation starts. This simple type's contents are a restriction of the xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 data type.

```

<xs:simpleType name="ST_AnimStart">
  <xs:restriction base="xs:string">
    <xs:enumeration value="c"/>
    <xs:enumeration value="w"/>
    <xs:enumeration value="a"/>
  </xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for ST_AnimStart:

Value	Meaning
c	The animation starts on click.
w	The animation starts when the previous animation in the animation sequence starts.
a	The animation starts after the previous animation in the animation sequence ends.

2.2.5.11 ST_EffectType

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A simple type that specifies the type of animation effect for a shape. This simple type's contents are a restriction of the xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 data type.

```

<xs:simpleType name="ST_EffectType">
  <xs:restriction base="xs:string">

```

```

    <xs:enumeration value="entr"/>
    <xs:enumeration value="exit"/>
    <xs:enumeration value="emph"/>
    <xs:enumeration value="path"/>
  </xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for ST_EffectType:

Value	Meaning
entr	The animation is an entrance animation.
exit	The animation is an exit animation.
emph	The animation is an emphasis animation.
path	The animation is a motion path animation.

2.2.5.12 ST_HideAfterAnimType

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared

A simple type that specifies if and when an animated shape is hidden after the animation effect ends. This simple type's content are a restriction of the xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 data type.

```

<xs:simpleType name="ST_HideAfterAnimType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="non"/>
    <xs:enumeration value="imm"/>
    <xs:enumeration value="clk"/>
  </xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for ST_HideAfterAnimType:

Value	Meaning
non	Do not hide the object after the animation effect ends.
imm	Hide the object immediately after the animation effect ends.
clk	Hide the object on the click following the animation effect.

2.2.6 Attributes

None.

2.2.7 Groups

None.

2.2.8 Attribute Groups

None.

2.3 Directory Service Schema Elements

None.

3 Protocol Details

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

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

Except where specified, protocol clients SHOULD interpret HTTP status codes returned by the protocol server as specified in [\[RFC2616\]](#) (Section 10, Status Code Definitions).

This protocol allows protocol servers to notify protocol clients of application-level faults using SOAP faults. Except where specified, these SOAP faults are not significant for interoperability, and protocol clients can interpret them in an implementation-specific manner.

This protocol allows protocol servers to perform implementation-specific authorization checks and notify protocol clients of authorization faults either using HTTP status codes or using SOAP faults as specified previously in this section.

3.1 Server Details

The following high-level sequence diagram illustrates the operation of the protocol.

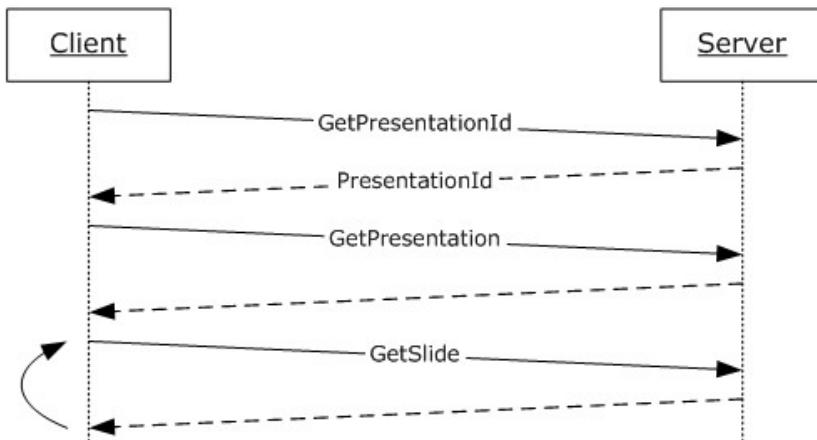


Figure 2: PowerPoint Web Viewer Presentation Data Protocol high-level sequence diagram.

First, a protocol client sends the **IPptService_GetPresentationId_InputMessage SOAP message** as specified in [\[MS-PWEDPS\]](#) section 3.1.4.10 to obtain a valid **presentationId** within an **IPptService_GetPresentationId_OutputMessage**. Next, a protocol client sends an [IPptViewingService_GetPresentation_InputMessage](#) message and the protocol server responds with an [IPptViewingService_GetPresentation_OutputMessage](#) containing information about the requested presentation. Next, the protocol client sends one or more [IPptViewingService_GetSlide_InputMessage](#) messages and the protocol server responds with an

[IPptViewingService_GetSlide_OutputMessage](#) containing information about the requested presentation slide.

The following high-level sequence diagram illustrates the operation of the attendee client protocol.

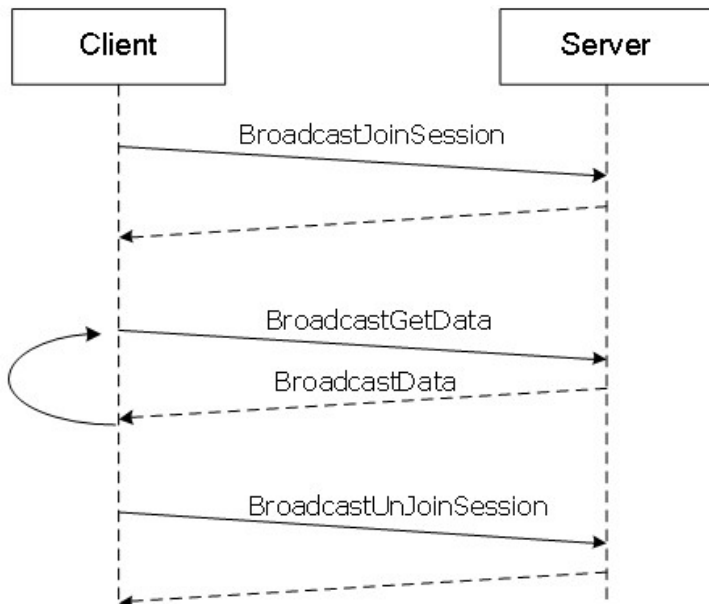


Figure 3: PowerPoint Web Viewer Presentation Data Protocol high-level sequence diagram for attendee clients.

First, a protocol client acting as slide show broadcast attendee sends a [BroadcastJoinSession](#) message and the protocol server responds with a [BroadcastJoinSessionResponse](#) message to acknowledge the request. Next, the protocol client sends one or more [BroadcastGetData](#) messages and the server sends a [BroadcastGetDataResponse](#) message containing the current data about the state of a broadcast. Finally, the protocol client SHOULD <1> send a [BroadcastUnjoinSession](#) message to unjoin from the session and the protocol server will respond with a [BroadcastUnjoinSessionResponse](#) message.

To obtain a valid **presentationId** for use with the protocol methods [GetPresentation](#) and [GetSlide](#), a protocol client acting as a slide show broadcast attendee sends a **GetMetadataXml** request as specified in [\[MS-OMPWHP\]](#) section 3.1.4.1.1. A valid **presentationId** is the value of the query string parameter named "presentationId" in the **url** attribute of **slideImage** element in the **mobilePres** root element returned by the **GetMetadataXml** method, as specified in [\[MS-PWVRSC\]](#) section 2.3.10.

3.1.1 Abstract Data Model

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

presentationId: An entity which represents a unique identifier for a presentation.

SessionId: An entity which represents a unique identifier for a broadcast session.

3.1.2 Timers

None.

3.1.3 Initialization

The protocol server MUST expose its Web methods at the following URL, which builds upon a base URL. The URL MUST conform to the following structure: *base URL/_vti_bin/ppt/view.svc*. This is the minimal required structure. Case-sensitivity is specific to the protocol server implementation.

3.1.4 Message Processing Events and Sequencing Rules

Section [3.1](#) specifies the sequencing of the protocol messages and how they relate to each other. The following sections specify the details of each individual message.

This specification includes the following **WSDL operations**:

WSDL Operation	Description
BinaryConvert	The BinaryConvert operation is used to convert a presentation from the binary file format for a PPT file (.ppt) described in [MS-PPT] to the Office Open XML File Format Specification PresentationML format described in the [ISO/IEC-29500:2008] specification.
BroadcastGetData	The BroadcastGetData operation is used by the broadcast attendee to retrieve information about the current state of the broadcast on the protocol server.
BroadcastJoinSession	The BroadcastJoinSession operation is used to join an attendee client to a slide show broadcast session.
BroadcastUnjoinSession	The BroadcastUnjoinSession operation SHOULD <2> be used to unjoin an attendee client from a slide show broadcast session.
GetNotesHtmlServiceResult	The GetNotesHtmlServiceResult protocol method returns the presentation notes content of a specified presentation.
GetPresentation	The GetPresentation protocol method returns information about the content of a specified presentation.
GetPresentationInfoXml	Reserved. MUST not be called.
GetSlide	The GetSlide protocol method returns information about the content of a specified presentation slide contained within a specified presentation.
GetSlideInfo	Reserved. MUST not be called.

3.1.4.1 BinaryConvert

The **BinaryConvert** operation is used to convert a presentation from the binary file format for a PPT file (.ppt) described in [\[MS-PPT\]](#) to the Office Open XML File Format Specification PresentationML format described in the [\[ISO/IEC-29500:2008\]](#) specification.

```
<wsdl:operation name="BinaryConvert">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
    ingService/IPptViewingService/BinaryConvert"
```

```

message="tns:IPptViewingService_BinaryConvert_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsd1"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BinaryConvertResponse"
message="tns:IPptViewingService_BinaryConvert_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsd1"/>
</wsdl:operation>

```

The protocol client sends an **IPptViewingService_BinaryConvert_InputMessage** request message, and the protocol server MUST respond with an **IPptViewingService_BinaryConvert_OutputMessage** response message as follows:

3.1.4.1.1 Messages

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

3.1.4.1.1.1 IPptViewingService_BinaryConvert_InputMessage

The request WSDL message for the **BinaryConvert** WSDL operation.

The **SOAP action** value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IP
ptViewingService/BinaryConvert

```

The **SOAP body** contains the **BinaryConvert** element.

3.1.4.1.1.2 IPptViewingService_BinaryConvert_OutputMessage

The response WSDL message for the **BinaryConvert** WSDL operation.

The SOAP body contains the **BinaryConvertResponse** element.

3.1.4.1.2 Elements

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

3.1.4.1.2.1 BinaryConvert

The input data for the **BinaryConvert** WSDL operation.

```

<xs:element name="BinaryConvert">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

presentationId: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the identifier of the presentation to convert, as specified in [\[MS-PWEDPS\]](#) section 3.1.4.10. This element MUST be present.

3.1.4.1.2.2 BinaryConvertResponse

The result data for the **BinaryConvert** WSDL operation.

```
<xs:element name="BinaryConvertResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:q13="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="BinaryConvertResult" nillable="true" type="q13:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

BinaryConvertResult: A [ServiceResult](#) element that specifies the result of the operation. If the **Result** child element of **BinaryConvertResult** is present, then the **Result** child element MUST be a [PptViewingService.BinaryConversionResult](#).

3.1.4.1.3 Complex Types

None.

3.1.4.1.4 Simple Types

None.

3.1.4.1.5 Attributes

None.

3.1.4.1.6 Groups

None.

3.1.4.1.7 Attribute Groups

None.

3.1.4.2 BroadcastGetData

The **BroadcastGetData** operation is used by the broadcast attendee to retrieve information about the current state of the broadcast on the protocol server.

```
<wsdl:operation name="BroadcastGetData">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BroadcastGetData"
message="tns:IPptViewingService_BroadcastGetData_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BroadcastGetDataResponse"
message="tns:IPptViewingService_BroadcastGetData_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends a **BroadcastGetDataSoapIn** request message, and the protocol server MUST respond with a **BroadcastGetDataSoapOut** response message as follows:

3.1.4.2.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.2.1.1 IPptViewingService_BroadcastGetData_InputMessage

The request WSDL message for the **BroadcastGetData** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IPptViewingService/BroadcastGetData
```

The SOAP body contains the **BroadcastGetData** element.

3.1.4.2.1.2 IPptViewingService_BroadcastGetData_OutputMessage

The response WSDL message for the **BroadcastGetData** WSDL operation.

The SOAP body contains the **BroadcastGetDataResponse** element.

3.1.4.2.2 Elements

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

3.1.4.2.2.1 BroadcastGetData

The input data for the **BroadcastGetData** WSDL operation.

```
<xs:element name="BroadcastGetData">
  <xs:complexType>
    <xs:sequence>
      <xs:element
        xmlns:q11="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data" minOccurs="0" name="user" nillable="true" type="q11:BroadcastUser"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

user: A [BroadcastUser](#) element that is obtained by making a [BroadcastJoinSession](#) web method call. This element MUST be present.

3.1.4.2.2.2 BroadcastGetDataResponse

The result data for the **BroadcastGetData** WSDL operation.

```
<xs:element name="BroadcastGetDataResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:q12="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
        name="BroadcastGetDataResult" nillable="true" type="q12:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
</xs:sequence>
</xs:complexType>
</xs:element>
```

BroadcastGetDataResult: A [ServiceResult](#) that specifies the result of the operation. This element MUST be present. The **Result** child element MUST be a [BroadcastData](#) element if the **Error** child element is not present.

3.1.4.2.3 Complex Types

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

3.1.4.2.3.1 BroadcastUser

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data>

A complex type that specifies a user of a broadcast session.

```
<xs:complexType name="BroadcastUser">
  <xs:sequence>
    <xs:element minOccurs="0" name="SessionId" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="UserToken" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

SessionId: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the identifier of the broadcast session on the protocol server. This element MUST be present.

UserToken: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the identifier of a user of the broadcast session on the protocol server. This element MUST be present.

3.1.4.2.4 Simple Types

None.

3.1.4.2.5 Attributes

None.

3.1.4.2.6 Groups

None.

3.1.4.2.7 Attribute Groups

None.

3.1.4.3 BroadcastJoinSession

The **BroadcastJoinSession** operation is used to join an attendee client to a slide show broadcast session.


```

<wsdl:operation name="BroadcastJoinSession">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BroadcastJoinSession"
message="tns:IPptViewingService_BroadcastJoinSession_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BroadcastJoinSessionResponse"
message="tns:IPptViewingService_BroadcastJoinSession_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends a **BroadcastJoinSessionSoapIn** request message, and the protocol server MUST respond with a **BroadcastJoinSessionSoapOut** response message as follows:

3.1.4.3.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.3.1.1 IPptViewingService_BroadcastJoinSession_InputMessage

The request WSDL message for the **BroadcastJoinSession** WSDL operation.

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IP
ptViewingService/BroadcastJoinSession

```

The SOAP body contains the **BroadcastJoinSession** element.

3.1.4.3.1.2 IPptViewingService_BroadcastJoinSession_OutputMessage

The response WSDL message for the **BroadcastJoinSession** WSDL operation.

The SOAP body contains the **BroadcastJoinSessionResponse** element.

3.1.4.3.2 Elements

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

3.1.4.3.2.1 BroadcastJoinSession

The input data for the **BroadcastJoinSession** WSDL operation.

```

<xs:element name="BroadcastJoinSession">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="sessionId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

sessionId: A xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the broadcast session to join. This element **MUST** be present.

3.1.4.3.2 BroadcastJoinSessionResponse

The result data for the **BroadcastJoinSession** WSDL operation.

```
<xs:element name="BroadcastJoinSessionResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:q9="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="BroadcastJoinSessionResult" nillable="true" type="q9:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

BroadcastJoinSessionResult: A [ServiceResult](#) that specifies the result of the operation. This element **MUST** be present. The **Result** child element **MUST** be a [BroadcastUser](#) if the **Error** child element is not present.

3.1.4.3.3 Complex Types

None.

3.1.4.3.4 Simple Types

None.

3.1.4.3.5 Attributes

None.

3.1.4.3.6 Groups

None.

3.1.4.3.7 Attribute Groups

None.

3.1.4.4 BroadcastUnjoinSession

The **BroadcastUnjoinSession** operation **SHOULD** [<3>](#) be used to unjoin an attendee client from a slide show broadcast session.

```
<wsdl:operation name="BroadcastUnjoinSession">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BroadcastUnjoinSession"
message="tns:IPptViewingService_BroadcastUnjoinSession_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BroadcastUnjoinSessionResponse"
```

```
message="tns:IPptViewingService_BroadcastUnjoinSession_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl" />
</wsdl:operation>
```

The protocol client sends a **BroadcastUnjoinSessionSoapIn** request message, and the protocol server MUST respond with a **BroadcastUnjoinSessionSoapOut** response message as follows:

3.1.4.4.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.4.1.1 IPptViewingService_BroadcastUnjoinSession_InputMessage

The request WSDL message for the **BroadcastUnjoinSession** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IPptViewingService/BroadcastUnjoinSession
```

The SOAP body contains the **BroadcastUnjoinSession** element.

3.1.4.4.1.2 IPptViewingService_BroadcastUnjoinSession_OutputMessage

The response WSDL message for the **BroadcastUnjoinSession** WSDL operation.

The SOAP body contains the **BroadcastUnjoinSessionResponse** element.

3.1.4.4.2 Elements

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

3.1.4.4.2.1 BroadcastUnjoinSession

The input data for the **BroadcastUnjoinSession** WSDL operation.

```
<xs:element name="BroadcastUnjoinSession">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="sessionId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

sessionId: A xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the broadcast session to unjoin from. This element MUST be present.

3.1.4.4.2.2 BroadcastUnjoinSessionResponse

The result data for the **BroadcastUnjoinSession** WSDL operation.

```
<xs:element name="BroadcastUnjoinSessionResponse">
  <xs:complexType>
```

```

    <xs:sequence>
      <xs:element xmlns:q10="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="BroadcastUnjoinSessionResult" nillable="true" type="q10:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

BroadcastUnjoinSessionResult: A [ServiceResult](#) that specifies the result of the operation. This element **MUST** be present. If the **Result** child element is present, it **MUST** be ignored by the protocol client. The protocol client **MAY** retry the request or display the error to the user if the **Error** child element is present.

3.1.4.4.3 Complex Types

None.

3.1.4.4.4 Simple Types

None.

3.1.4.4.5 Attributes

None.

3.1.4.4.6 Groups

None.

3.1.4.4.7 Attribute Groups

None.

3.1.4.5 GetNotesHtmlServiceResult

The **GetNotesHtmlServiceResult** protocol method returns the presentation notes content of a specified presentation.

```

<wsdl:operation name="GetNotesHtmlServiceResult">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetNotesHtmlServiceResult"
message="tns:IPptViewingService_GetNotesHtmlServiceResult_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetNotesHtmlServiceResultResponse"
message="tns:IPptViewingService_GetNotesHtmlServiceResult_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptViewingService_GetNotesHtmlServiceResult_InputMessage** request message and the protocol server **MUST** respond with an **IPptViewingService_GetNotesHtmlServiceResult_OutputMessage** response message as follows:

3.1.4.5.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.5.1.1 IPptViewingService_GetNotesHtmlServiceResult_InputMessage

The request WSDL message for the **GetNotesHtmlServiceResult** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IPptViewingService/GetNotesHtmlServiceResult
```

The SOAP body contains the **GetNotesHtmlServiceResult** element.

3.1.4.5.1.2 IPptViewingService_GetNotesHtmlServiceResult_OutputMessage

The response WSDL message for the **GetNotesHtmlServiceResult** WSDL operation.

The SOAP body contains the **GetNotesHtmlServiceResultResponse** element.

3.1.4.5.2 Elements

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

3.1.4.5.2.1 GetNotesHtmlServiceResult

The input data for the **GetNotesHtmlServiceResult** WSDL operation.

```
<xs:element name="GetNotesHtmlServiceResult">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="notesResourceId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element
        xmlns:q7="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared" minOccurs="0" name="powerpointView" type="q7:PowerpointView"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

notesResourceId: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the identifier of the presentation notes to be retrieved. This element **MUST** be present and **MUST** be a valid identifier contained within the **notesId** child element of a [CT_Slide](#) retrieved with the same **presentationId** identifier.

presentationId: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the identifier of the presentation containing the presentation notes to be retrieved, as specified in [\[MS-PWEDPS\]](#) section 3.1.4.12. This element **MUST** be present.

powerpointView: A [q7:PowerpointView](#) element that specifies the view for which notes are being requested. The value **MUST** be **OutlineView** or **ReadingView**.

3.1.4.5.2 GetNotesHtmlServiceResultResponse

The result data for the **GetNotesHtmlServiceResult** WSDL operation.

```
<xs:element name="GetNotesHtmlServiceResultResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:q8="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetNotesHtmlServiceResultResult" nillable="true" type="q8:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetNotesHtmlServiceResultResult: A [ServiceResult](#) that specifies the result of the operation. The **Result** child element MUST be an xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 that contains a valid [\[HTML\]](#) fragment.

3.1.4.5.3 Complex Types

None.

3.1.4.5.4 Simple Types

None.

3.1.4.5.5 Attributes

None.

3.1.4.5.6 Groups

None.

3.1.4.5.7 Attribute Groups

None.

3.1.4.6 GetPresentation

The **GetPresentation** protocol method returns information about the content of a specified presentation.

```
<wsdl:operation name="GetPresentation">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetPresentation"
message="tns:IPptViewingService_GetPresentation_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetPresentationResponse"
message="tns:IPptViewingService_GetPresentation_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptViewingService_GetPresentation_InputMessage** request message and the protocol server MUST respond with an **IPptViewingService_GetPresentation_OutputMessage** response message as follows:

3.1.4.6.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.6.1.1 IPptViewingService_GetPresentation_InputMessage

The request WSDL message for the **GetPresentation** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IPptViewingService/GetPresentation
```

The SOAP body contains the **GetPresentation** element.

3.1.4.6.1.2 IPptViewingService_GetPresentation_OutputMessage

The response WSDL message for the **GetPresentation** WSDL operation.

The SOAP body contains the **GetPresentationResponse** element.

3.1.4.6.2 Elements

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

3.1.4.6.2.1 GetPresentation

The input data for the **GetPresentation** WSDL operation.

```
<xs:element name="GetPresentation">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element
        xmlns:q2="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared"
        minOccurs="0" name="powerpointView" type="q2:PowerpointView"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the identifier of the presentation to retrieve information about, as specified in [\[MS-PWEDPS\]](#) section 3.1.4.10. This element MUST be present.

powerpointView: A [PowerpointView](#) element that specifies the view. This element MUST be present.

3.1.4.6.2.2 GetPresentationResponse

The result data for the **GetPresentation** WSDL operation.

```

<xs:element name="GetPresentationResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:q3="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetPresentationResult" nillable="true" type="q3:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

GetPresentationResult: A [ServiceResult](#) element that specifies the result of the operation. If the **Result** child element of **GetPresentationResult** is present, then the **Result** child element MUST be a [CT_PresentationShowInfo](#) element.

3.1.4.6.3 Complex Types

None.

3.1.4.6.4 Simple Types

None.

3.1.4.6.5 Attributes

None.

3.1.4.6.6 Groups

None.

3.1.4.6.7 Attribute Groups

None.

3.1.4.7 GetPresentationInfoXml

Reserved. MUST not be called.

```

<wsdl:operation name="GetPresentationInfoXml">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetPresentationInfoXml"
message="tns:IPptViewingService_GetPresentationInfoXml_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetPresentationInfoXmlResponse"
message="tns:IPptViewingService_GetPresentationInfoXml_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

Reserved. MUST not be called.

3.1.4.7.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.7.1.1 IPptViewingService_GetPresentationInfoXml_InputMessage

The request WSDL message for the **GetPresentationInfoXml** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IPptViewingService/GetPresentationInfoXml
```

The SOAP body contains the **GetPresentationInfoXml** element.

3.1.4.7.1.2 IPptViewingService_GetPresentationInfoXml_OutputMessage

The response WSDL message for the **GetPresentationInfoXml** WSDL operation.

The SOAP body contains the **GetPresentationInfoXmlResponse** element.

3.1.4.7.2 Elements

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

3.1.4.7.2.1 GetPresentationInfoXml

The input data for the **GetPresentationInfoXml** WSDL operation.

```
<xs:element name="GetPresentationInfoXml">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="powerpointView" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: Reserved. MUST be ignored.

powerpointView: Reserved. MUST be ignored.

3.1.4.7.2.2 GetPresentationInfoXmlResponse

The result data for the **GetPresentationInfoXml** WSDL operation.

```
<xs:element name="GetPresentationInfoXmlResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:q1="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetPresentationInfoXmlResult" nillable="true" type="q1:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetPresentationInfoXmlResult: Reserved. MUST be ignored.

3.1.4.7.3 Complex Types

None.

3.1.4.7.4 Simple Types

None.

3.1.4.7.5 Attributes

None.

3.1.4.7.6 Groups

None.

3.1.4.7.7 Attribute Groups

None.

3.1.4.8 GetSlide

The **GetSlide** protocol method returns information about the content of a specified presentation slide contained within a specified presentation.

```
<wsdl:operation name="GetSlide">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IPptViewingService/GetSlide"
    message="tns:IPptViewingService_GetSlide_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IPptViewingService/GetSlideResponse"
    message="tns:IPptViewingService_GetSlide_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptViewingService_GetSlide_InputMessage** request message and the protocol server MUST respond with an **IPptViewingService_GetSlide_OutputMessage** response message as follows:

3.1.4.8.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.8.1.1 IPptViewingService_GetSlide_InputMessage

The request WSDL message for the **GetSlide** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IPptViewingService/GetSlide
```

The SOAP body contains the **GetSlide** element.

3.1.4.8.1.2 IPptViewingService_GetSlide_OutputMessage

The response WSDL message for the **GetSlide** WSDL operation.

The SOAP body contains the **GetSlideResponse** element.

3.1.4.8.2 Elements

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

3.1.4.8.2.1 GetSlide

The input data for the **GetSlide** WSDL operation.

```
<xs:element name="GetSlide">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="slideInfoId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element
        xmlns:q5="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared" minOccurs="0" name="powerpointView" type="q5:PowerpointView"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

slideInfoId: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the identifier of the presentation slide to retrieve information about. This element MUST contain a string as specified in the **sldInfo** child element of the [CT_SlideEntry](#) element.

presentationId: An xs:string [\[XMLSCHEMA2\]](#) section 3.2.1 element that specifies the identifier of the presentation to retrieve information about, as specified in [\[MS-PWEDPS\]](#) section 3.1.4.10. This element MUST be present.

powerpointView: A [PowerpointView](#) element that specifies the view. This element MUST be present.

3.1.4.8.2.2 GetSlideResponse

The result data for the **GetSlide** WSDL operation.

```
<xs:element name="GetSlideResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:q6="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
        name="GetSlideResult" nillable="true" type="q6:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetSlideResult: A [ServiceResult](#) that specifies the result of the operation. If the **Result** child element of **GetSlideResult** is present, then the **Result** child element MUST be a [CT_SlideShowInfo](#).

3.1.4.8.3 Complex Types

None.

3.1.4.8.4 Simple Types

None.

3.1.4.8.5 Attributes

None.

3.1.4.8.6 Groups

None.

3.1.4.8.7 Attribute Groups

None.

3.1.4.9 GetSlideInfo

Reserved. MUST not be called.

```
<wsdl:operation name="GetSlideInfo">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IPptViewingService/GetSlideInfo"
    message="tns:IPptViewingService_GetSlideInfo_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IPptViewingService/GetSlideInfoResponse"
    message="tns:IPptViewingService_GetSlideInfo_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

Reserved. MUST not be called.

3.1.4.9.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.9.1.1 IPptViewingService_GetSlideInfo_InputMessage

The request WSDL message for the **GetSlideInfo** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/IPptViewingService/GetSlideInfo
```

The SOAP body contains the **GetSlideInfo** element.

3.1.4.9.1.2 IPptViewingService_GetSlideInfo_OutputMessage

The response WSDL message for the **GetSlideInfo** WSDL operation.

The SOAP body contains the **GetSlideInfoResponse** element.

3.1.4.9.2 Elements

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

3.1.4.9.2.1 GetSlideInfo

The input data for the **GetSlideInfo** WSDL operation.

```
<xs:element name="GetSlideInfo">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="slideInfoId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="resourceInfoListId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="powerpointView" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

slideInfoId: Reserved. MUST be ignored.

resourceInfoListId: Reserved. MUST be ignored.

presentationId: Reserved. MUST be ignored.

powerpointView: Reserved. MUST be ignored.

3.1.4.9.2.2 GetSlideInfoResponse

The result data for the **GetSlideInfo** WSDL operation.

```
<xs:element name="GetSlideInfoResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:q4="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
        name="GetSlideInfoResult" nillable="true" type="q4:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetSlideInfoResult: Reserved. MUST be ignored.

3.1.4.9.3 Complex Types

None.

3.1.4.9.4 Simple Types

None.

3.1.4.9.5 Attributes

None.

3.1.4.9.6 Groups

None.

3.1.4.9.7 Attribute Groups

None.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

3.2 Client Details

None.

3.2.1 Abstract Data Model

None.

3.2.2 Timers

None.

3.2.3 Initialization

None.

3.2.4 Message Processing Events and Sequencing Rules

None.

3.2.5 Timer Events

None.

3.2.6 Other Local Events

None.

4 Protocol Examples

The following example contains a sample interaction between the protocol client and the protocol server.

4.1 Interaction Between Protocol Client and Protocol Server

The protocol client sends a request to the PowerPoint Web Editor Data Protocol [\[MS-PWEDPS\]](#) server for the identifier of a presentation. The following **IPptService_GetPresentationId_InputMessage** message is sent to the protocol server:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetPresentationId
      xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServer
      InternalService/">
      <rawDocId>/Shared Documents/MyFile.pptx</rawDocId>
    </GetPresentationId>
  </s:Body>
</s:Envelope>
```

The protocol server responds with the appropriate identifier for the presentation. The following **IPptService_GetPresentationId_OutputMessage** message is sent to the protocol client:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetPresentationIdResponse
      xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServer
      InternalService/">
      <GetPresentationIdResult>d=F39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a9a007e0723241f
      m18af359c163142ad81aa5449d829efd1m</GetPresentationIdResult>
    </GetPresentationIdResponse>
  </s:Body>
</s:Envelope>
```

Next, the protocol client sends a request to the protocol server for information about the presentation. The following **IPptViewingService_GetPresentation_InputMessage** message is sent to the protocol server:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetPresentation
      xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingSer
      vice/"><presentationId>d=F39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a9a007e0723241fm1
      8af359c163142ad81aa5449d829efd1m</presentationId>
      <powerpointView>ReadingView</powerpointView>
    </GetPresentation>
  </s:Body>
</s:Envelope>
```

The protocol server responds with the appropriate information about the presentation. The following **IPptViewingService_GetPresentation_OutputMessage** message is sent to the protocol client:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetPresentationResponse
      xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/">
      <GetPresentationResult xmlns:a="http://schemas.datacontract.org/2004/07/p"
        xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Error i:nil="true"/>
        <a:Result i:type="b:CT_PresentationShowInfo"
          xmlns:b="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared">
          <b:h>600</b:h>
          <b:hasCmt>>false</b:hasCmt>
          <b:rtl>>false</b:rtl>
          <b:schmVer>0</b:schmVer>
          <b:showPr>
            <b:Item i:type="b:CT_EmptyType"/>
          </b:showPr>
          <b:sldLst>
            <b:CT_SlideEntry>
              <b:id>256</b:id>
              <b:sldInfo>100.sldInfo.xml</b:sldInfo>
              <b:title>Edit #1 Edit #4</b:title>
            </b:CT_SlideEntry>
            <b:CT_SlideEntry>
              <b:id>257</b:id>
              <b:sldInfo>101.sldInfo.xml</b:sldInfo>
              <b:title>Edit #2 </b:title>
            </b:CT_SlideEntry>
          </b:sldLst>
          <b:w>800</b:w>
        </a:Result>
      </GetPresentationResult>
    </GetPresentationResponse>
  </s:Body>
</s:Envelope>

```

Next, the protocol client sends a request to the protocol server for information about a presentation slide. The **slideInfoId** is specified by the **sldInfo** child element of the [CT_SlideEntry](#) element in the response of the previous request. The following **IPptViewingService_GetSlide_InputMessage** message is sent to the protocol server:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetSlide
      xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/">
      <slideInfoId>100.sldInfo.xml</slideInfoId>

      <presentationId>d=F39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a9a007e0723241fm18af359c163142ad81aa5449d829efd1m</presentationId>
      <powerpointView>ReadingView</powerpointView>
    </GetSlide>
  </s:Body>
</s:Envelope>

```


The protocol server responds with the appropriate information about the presentation slide. The following **IPptViewingService_GetSlide_OutputMessage** message is sent to the protocol client:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetSlideResponse
      xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/">
      <GetSlideResult xmlns:a="http://schemas.datacontract.org/2004/07/p"
        xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Error i:nil="true"/>
        <a:Result i:type="b:CT_SlideShowInfo"
          xmlns:b="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared">
          <b:errLst i:nil="true"/>
          <b:errMsgLst xmlns:c="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
          <b:schmVer>0</b:schmVer>
          <b:sld>
            <b:animLst>
              <b:anim i:nil="true"/>
              <b:id>0</b:id>
            </b:animLst>
            <b:animSeqs i:nil="true"/>
            <b:bg>
              <b:Item i:type="b:CT_SolidFill">
                <b:clr>///</b:clr>
                <b:transp>0</b:transp>
              </b:Item>
            </b:bg>
            <b:bgReg i:nil="true"/>
            <b:hidden>>false</b:hidden>
            <b:id>256</b:id>
            <b:layoutInfo i:nil="true"/>
            <b:notesId/>
            <b:resCat>
              <b:CT_ResourceEntry>
                <b:id>S0_1</b:id>
              </b:CT_ResourceEntry>
            </b:resCat>
            <b:name>./imagehandler.ashx?PV=0&Pid=d%3DF39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a9a007e0723241fm18af359c163142ad81aa5449d829efd1m&Rid=S0%5F1%2Epng</b:name>
            </b:CT_ResourceEntry>
          </b:resCat>
          <b:spLst>
            <b:CT_Shape>
              <b:bgSp>>false</b:bgSp>
              <b:children i:nil="true"/>
              <b:clckTgts i:nil="true"
                xmlns:c="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
              <b:id>3</b:id>
              <b:pos>
                <b:x>60</b:x>
                <b:y>186</b:y>
              </b:pos>
              <b:reg>
                <b:h>129</b:h>
                <b:imgID>S0_1</b:imgID>
                <b:w>680</b:w>
                <b:x>0</b:x>
                <b:y>0</b:y>
              </b:reg>
            </b:CT_Shape>
          </b:spLst>
        </a:Result>
      </GetSlideResult>
    </GetSlideResponse>
  </s:Body>
</s:Envelope>
```

```

        </b:CT_Shape>
    </b:spLst>
    <b:ssTimes i:nil="true"/>
    <b:title/>
    <b:trans i:nil="true"/>
</b:sld>
</a:Result>
</GetSlideResult>
</GetSlideResponse>
</s:Body>
</s:Envelope>

```

4.1.1 Interaction Between Attendee Protocol Client and Protocol Server

The attendee protocol client begins by sending a request to the protocol server to join the slide show broadcast. The following **BroadcastJoinSessionSoapIn** message is sent to the protocol server:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soap:Body>
    <BroadcastJoinSession
xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/">
      <sessionId>601022d2-306a-43fb-8ffd-a4a739cba8c0</sessionId>
    </BroadcastJoinSession>
  </soap:Body>
</soap:Envelope>

```

The protocol server responds with a message to acknowledge the request and to provide host information and a session identifier to be used by the attendee protocol client for future requests. The following **BroadcastJoinSessionSoapOut** message is sent to the attendee protocol client:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soap:Body>
    <BroadcastJoinSessionResult>
      <Result xsi:type="BroadcastUser">
        <SessionId>601022d2-306a-43fb-8ffd-a4a739cba8c0</SessionId>
        <UserToken>f52b7833-6167-438f-b71d-ea99219c4da3</UserToken>
      </Result>
    </BroadcastJoinSessionResult>
  </soap:Body>
</soap:Envelope>

```

Next, the attendee protocol client requests information about the current state of the slide show broadcast from the protocol server. The following **BroadcastGetDataSoapIn** message is sent to the protocol server:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soap:Body>
    <BroadcastGetData
xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService">

```

```

    <User>
      <SessionId>601022d2-306a-43fb-8ffd-a4a739cba8c0</SessionId>
      <UserToken>f52b7833-6167-438f-b71d-ea99219c4da3</UserToken>
    </User>
  </BroadcastGetData>
</soap:Body>
</soap:Envelope>

```

The protocol server responds with information about the current state of the slide show broadcast. The following **BroadcastGetDataSoapOut** message is sent to the attendee protocol client:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soap:Body>
    <BroadcastGetDataResponse>
      <BroadcastGetDataResult>
        <Result xsi:type="BroadcastData">
          <SlideShowState>Normal</SlideShowState>
          <HostToken> /sites/broadcast/9d51d0b1f4774b6893cb728c0ba15a57/faf48352-f17f-
46ed-ad44-9adac3fd37bb.pptx</HostToken>
          <SlideId>256</SlideId>
          <AnimationStepDataList>
            <BroadcastAnimationStepData>
              <TimelineId>0_anim</TimelineId>
              <Step>0</Step>
            </BroadcastAnimationStepData>
          </AnimationStepDataList>
          <SequenceNumber>2</SequenceNumber>
        </Result>
      </BroadcastGetDataResult>
    </BroadcastGetDataResponse>
  </soap:Body>
</soap:Envelope>

```

The attendee protocol client continues to send **BroadcastGetDataSoapIn** messages to request the latest state of the slide show broadcast.

Finally, the attendee protocol client unjoins from the slide show broadcast by sending **BroadcastUnjoinSessionSoapIn** message to the protocol server:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soap:Body>
    <BroadcastUnjoinSession
xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/">
      <sessionId>601022d2-306a-43fb-8ffd-a4a739cba8c0</sessionId>
    </BroadcastUnjoinSession>
  </soap:Body>
</soap:Envelope>

```

The protocol server responds by sending the **BroadcastUnjoinSessionSoapOut** message to the attendee protocol client:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

```

```
<soap:Body>
  <BroadcastUnjoinSessionResponse
xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/">
    <BroadcastUnjoinSessionResult xmlns:a="http://schemas.datacontract.org/2004/07/p"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
      <a>Error i:nil="true"/><a:Result i:nil="true"/>
    </BroadcastUnjoinSessionResult>
  </BroadcastUnjoinSessionResponse>
</soap:Body>
</soap:Envelope>
```

5 Security

5.1 Security Considerations for Implementers

A [GetNotesHtmlServiceResultResponse](#) contains [\[HTML\]](#) fragments. HTML fragments are susceptible to **cross-site scripting** attacks. Implementers are advised to sanitize these HTML fragments when generating or processing a [GetNotesHtmlServiceResultResponse](#).

There are no additional security considerations that are specific to this protocol. General security considerations pertaining to [\[RFC2822\]](#) apply.

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

For ease of implementation, the full WSDL is provided:

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:i0="http://tempuri.org/"
xmlns:wsa10="http://www.w3.org/2005/08/addressing"
xmlns:tns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
xmlns:wsx="http://schemas.xmlsoap.org/ws/2004/09/mex"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsd1"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
xmlns:wsap="http://schemas.xmlsoap.org/ws/2004/08/addressing/policy"
xmlns:misc="http://schemas.microsoft.com/ws/2005/12/wsd1/contract"
xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
xmlns:soap="http://schemas.xmlsoap.org/wsd1/soap/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://schemas.xmlsoap.org/wsd1/soap12/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
name="PptViewingService"
targetNamespace="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/"
xmlns:wsd1="http://schemas.xmlsoap.org/wsd1/">
  <wsdl:import namespace="http://tempuri.org/" />
  <wsdl:types>
    <xs:schema elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointViewingService/"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:element name="GetPresentationInfoXml">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string" />
            <xs:element minOccurs="0" name="powerpointView" nillable="true" type="xs:string" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="GetPresentationInfoXmlResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element xmlns:q1="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetPresentationInfoXmlResult" nillable="true" type="q1:ServiceResult" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="GetPresentation">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string" />
            <xs:element
xmlns:q2="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared"
minOccurs="0" name="powerpointView" type="q2:PowerpointView" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="GetPresentationResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element xmlns:q3="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetPresentationResult" nillable="true" type="q3:ServiceResult" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:schema>
  </wsdl:types>

```

```

        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="GetSlideInfo">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="slideInfoId" nillable="true" type="xs:string" />
            <xs:element minOccurs="0" name="resourceInfoListId" nillable="true"
type="xs:string" />
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"
/>
            <xs:element minOccurs="0" name="powerpointView" nillable="true" type="xs:string"
/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="GetSlideInfoResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:q4="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetSlideInfoResult" nillable="true" type="q4:ServiceResult" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="GetSlide">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="slideInfoId" nillable="true" type="xs:string" />
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"
/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element
xmlns:q5="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interfac
e.Shared" minOccurs="0" name="powerpointView" type="q5:PowerpointView" />
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="GetSlideResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:q6="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetSlideResult" nillable="true" type="q6:ServiceResult" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="GetNotesHtmlServiceResult">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="notesResourceId" nillable="true" type="xs:string"
/>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"
/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element
xmlns:q7="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interfac
e.Shared" minOccurs="0" name="powerpointView" type="q7:PowerpointView" />
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="GetNotesHtmlServiceResultResponse">
    <xs:complexType>
        <xs:sequence>

```

```

        <xs:element xmlns:q8="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetNotesHtmlServiceResultResult" nillable="true" type="q8:ServiceResult" />
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="BroadcastJoinSession">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="sessionId" nillable="true" type="xs:string" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="BroadcastJoinSessionResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:q9="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="BroadcastJoinSessionResult" nillable="true" type="q9:ServiceResult" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="BroadcastGetData">
    <xs:complexType>
        <xs:sequence>
            <xs:element
xmlns:q10="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data" minOccurs="0" name="user" nillable="true" type="q10:BroadcastUser" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="BroadcastGetDataResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:q11="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="BroadcastGetDataResult" nillable="true" type="q11:ServiceResult" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="BinaryConvert">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"
/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="BinaryConvertResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:q12="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="BinaryConvertResult" nillable="true" type="q12:ServiceResult" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:schema>
<xs:schema xmlns:tns="http://schemas.datacontract.org/2004/07/p"
elementFormDefault="qualified" targetNamespace="http://schemas.datacontract.org/2004/07/p"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:complexType name="SlideInfoResult">
        <xs:sequence>

```



```

        <xs:element minOccurs="0" name="ResourceInfoList" nillable="true" type="xs:string"
/>
    <xs:element minOccurs="0" name="Slide" nillable="true" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:element name="SlideInfoResult" nillable="true" type="tns:SlideInfoResult" />
<xs:complexType name="PptViewingService.BinaryConversionResult">
  <xs:sequence>
    <xs:element minOccurs="0" name="EditUrl" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="FExists" type="xs:boolean" />
    <xs:element minOccurs="0" name="ViewUrl" nillable="true" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:element name="PptViewingService.BinaryConversionResult" nillable="true"
type="tns:PptViewingService.BinaryConversionResult" />
<xs:complexType name="ServiceResult">
  <xs:sequence>
    <xs:element minOccurs="0" name="Error" nillable="true" type="tns:ServiceError" />
    <xs:element minOccurs="0" name="Result" nillable="true" type="xs:anyType" />
  </xs:sequence>
</xs:complexType>
<xs:element name="ServiceResult" nillable="true" type="tns:ServiceResult" />
<xs:complexType name="ServiceError">
  <xs:sequence>
    <xs:element minOccurs="0" name="Message" nillable="true" type="xs:string" />
    <xs:element
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.Int
erface" minOccurs="0" name="RecommendedActions" type="q1:ClientActions" />
    <xs:element minOccurs="0" name="Title" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="Type" type="tns:ServiceErrorType" />
  </xs:sequence>
</xs:complexType>
<xs:element name="ServiceError" nillable="true" type="tns:ServiceError" />
<xs:simpleType name="ServiceErrorType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="UnknownError" />
    <xs:enumeration value="ApplicationError" />
    <xs:enumeration value="Timeout" />
    <xs:enumeration value="ServiceBusy" />
  </xs:restriction>
</xs:simpleType>
<xs:element name="ServiceErrorType" nillable="true" type="tns:ServiceErrorType" />
</xs:schema>
<xs:schema
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.In
terface" elementFormDefault="qualified"
targetNamespace="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.P
ipe.Interface" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="ClientActions">
    <xs:list>
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="None">
            <xs:annotation>
              <xs:appinfo>
                <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">0</EnumerationValue>
              </xs:appinfo>
            </xs:annotation>
          </xs:enumeration>

```

```

        <xs:enumeration value="Dismiss">
          <xs:annotation>
            <xs:appinfo>
              <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">1</EnumerationValue>
            </xs:appinfo>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="Close">
          <xs:annotation>
            <xs:appinfo>
              <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">2</EnumerationValue>
            </xs:appinfo>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="OpenInClient">
          <xs:annotation>
            <xs:appinfo>
              <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">4</EnumerationValue>
            </xs:appinfo>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="Refresh">
          <xs:annotation>
            <xs:appinfo>
              <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">8</EnumerationValue>
            </xs:appinfo>
          </xs:annotation>
        </xs:enumeration>
      </xs:restriction>
    </xs:simpleType>
  </xs:list>
</xs:simpleType>
<xs:element name="ClientActions" nillable="true" type="tns:ClientActions" />
</xs:schema>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2003/10/Serialization/"
attributeFormDefault="qualified" elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/2003/10/Serialization/"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="anyType" nillable="true" type="xs:anyType" />
  <xs:element name="anyURI" nillable="true" type="xs:anyURI" />
  <xs:element name="base64Binary" nillable="true" type="xs:base64Binary" />
  <xs:element name="boolean" nillable="true" type="xs:boolean" />
  <xs:element name="byte" nillable="true" type="xs:byte" />
  <xs:element name="dateTime" nillable="true" type="xs:dateTime" />
  <xs:element name="decimal" nillable="true" type="xs:decimal" />
  <xs:element name="double" nillable="true" type="xs:double" />
  <xs:element name="float" nillable="true" type="xs:float" />
  <xs:element name="int" nillable="true" type="xs:int" />
  <xs:element name="long" nillable="true" type="xs:long" />
  <xs:element name="QName" nillable="true" type="xs:QName" />
  <xs:element name="short" nillable="true" type="xs:short" />
  <xs:element name="string" nillable="true" type="xs:string" />
  <xs:element name="unsignedByte" nillable="true" type="xs:unsignedByte" />
  <xs:element name="unsignedInt" nillable="true" type="xs:unsignedInt" />
  <xs:element name="unsignedLong" nillable="true" type="xs:unsignedLong" />

```

```

<xs:element name="unsignedShort" nillable="true" type="xs:unsignedShort" />
<xs:element name="char" nillable="true" type="tns:char" />
<xs:simpleType name="char">
  <xs:restriction base="xs:int" />
</xs:simpleType>
<xs:element name="duration" nillable="true" type="tns:duration" />
<xs:simpleType name="duration">
  <xs:restriction base="xs:duration">
    <xs:pattern value="\-?P(\d*D)?(T(\d*H)?(\d*M)?(\d*(\.\d*)?S)?)?" />
    <xs:minInclusive value="-P10675199DT2H48M5.4775808S" />
    <xs:maxInclusive value="P10675199DT2H48M5.4775807S" />
  </xs:restriction>
</xs:simpleType>
<xs:element name="guid" nillable="true" type="tns:guid" />
<xs:simpleType name="guid">
  <xs:restriction base="xs:string">
    <xs:pattern value="[\da-fA-F]{8}-[\da-fA-F]{4}-[\da-fA-F]{4}-[\da-fA-F]{4}-[\da-fA-F]{12}" />
  </xs:restriction>
</xs:simpleType>
<xs:attribute name="FactoryType" type="xs:QName" />
<xs:attribute name="Id" type="xs:ID" />
<xs:attribute name="Ref" type="xs:IDREF" />
</xs:schema>
<xs:schema
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared" elementFormDefault="qualified"
targetNamespace="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="CT_SlideLayoutInfo">
    <xs:sequence>
      <xs:element name="fTitleLayout" type="xs:boolean" />
      <xs:element name="layoutId" type="xs:unsignedInt" />
      <xs:element name="masterId" type="xs:unsignedInt" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="CT_SlideLayoutInfo" nillable="true" type="tns:CT_SlideLayoutInfo" />
  <xs:complexType name="CT_LayoutPosition">
    <xs:sequence>
      <xs:element name="x" type="xs:int" />
      <xs:element name="y" type="xs:int" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="CT_LayoutPosition" nillable="true" type="tns:CT_LayoutPosition" />
  <xs:complexType name="CT_Error">
    <xs:sequence>
      <xs:element name="id" type="xs:unsignedInt" />
      <xs:element xmlns:q1="http://schemas.microsoft.com/2003/10/Serialization/Arrays" name="p" nillable="true" type="q1:ArrayOfstring" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="CT_Error" nillable="true" type="tns:CT_Error" />
  <xs:complexType name="CT_ErrorInfo">
    <xs:sequence>
      <xs:element name="err" nillable="true" type="tns:CT_Error" />
      <xs:element name="schmVer" type="xs:unsignedInt" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="CT_ErrorInfo" nillable="true" type="tns:CT_ErrorInfo" />

```

```

<xs:complexType name="CT_ResourceEntry">
  <xs:sequence>
    <xs:element name="id" nillable="true" type="xs:string" />
    <xs:element name="name" nillable="true" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_ResourceEntry" nillable="true" type="tns:CT_ResourceEntry" />
<xs:complexType name="CT_MotionPathMoveCmd">
  <xs:sequence>
    <xs:element name="x" type="xs:int" />
    <xs:element name="y" type="xs:int" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_MotionPathMoveCmd" nillable="true" type="tns:CT_MotionPathMoveCmd"
/>
<xs:complexType name="CT_MotionPathLineCmd">
  <xs:sequence>
    <xs:element name="x" type="xs:int" />
    <xs:element name="y" type="xs:int" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_MotionPathLineCmd" nillable="true" type="tns:CT_MotionPathLineCmd"
/>
<xs:complexType name="CT_MotionPathCurveCmd">
  <xs:sequence>
    <xs:element name="x1" type="xs:int" />
    <xs:element name="x2" type="xs:int" />
    <xs:element name="x3" type="xs:int" />
    <xs:element name="y1" type="xs:int" />
    <xs:element name="y2" type="xs:int" />
    <xs:element name="y3" type="xs:int" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_MotionPathCurveCmd" nillable="true"
type="tns:CT_MotionPathCurveCmd" />
<xs:complexType name="CT_MotionPath">
  <xs:sequence>
    <xs:element xmlns:q2="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
name="Items" nillable="true" type="q2:ArrayOfanyType" />
    <xs:element name="rev" type="xs:boolean" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_MotionPath" nillable="true" type="tns:CT_MotionPath" />
<xs:complexType name="CT_Rectangle">
  <xs:sequence>
    <xs:element name="h" type="xs:int" />
    <xs:element name="l" type="xs:int" />
    <xs:element name="t" type="xs:int" />
    <xs:element name="w" type="xs:int" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_Rectangle" nillable="true" type="tns:CT_Rectangle" />
<xs:complexType name="CT_Hyperlink">
  <xs:sequence>
    <xs:element name="rct" nillable="true" type="tns:CT_Rectangle" />
    <xs:element name="tip" nillable="true" type="xs:string" />
    <xs:element name="url" nillable="true" type="xs:string" />
  </xs:sequence>
</xs:complexType>

```

```

<xs:element name="CT_Hyperlink" nillable="true" type="tns:CT_Hyperlink" />
<xs:complexType name="CT_Action">
  <xs:sequence>
    <xs:element name="rct" nillable="true" type="tns:CT_Rectangle" />
    <xs:element name="sldId" type="xs:unsignedInt" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_Action" nillable="true" type="tns:CT_Action" />
<xs:complexType name="CT_InteractiveAnimUse">
  <xs:sequence>
    <xs:element name="rct" nillable="true" type="tns:CT_Rectangle" />
    <xs:element name="seqId" type="xs:unsignedInt" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_InteractiveAnimUse" nillable="true"
type="tns:CT_InteractiveAnimUse" />
<xs:complexType name="CT_AnimTiming">
  <xs:sequence>
    <xs:element name="off" type="xs:unsignedInt" />
    <xs:element name="spd" nillable="true" type="xs:string" />
    <xs:element name="st" type="tns:ST_AnimStart" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_AnimTiming" nillable="true" type="tns:CT_AnimTiming" />
<xs:simpleType name="ST_AnimStart">
  <xs:restriction base="xs:string">
    <xs:enumeration value="c" />
    <xs:enumeration value="w" />
    <xs:enumeration value="a" />
  </xs:restriction>
</xs:simpleType>
<xs:element name="ST_AnimStart" nillable="true" type="tns:ST_AnimStart" />
<xs:complexType name="CT_AnimIteration">
  <xs:sequence>
    <xs:element name="interval" type="xs:int" />
    <xs:element name="reversed" type="xs:boolean" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_AnimIteration" nillable="true" type="tns:CT_AnimIteration" />
<xs:complexType name="CT_GenericImageRegion">
  <xs:sequence>
    <xs:element name="h" type="xs:int" />
    <xs:element name="imgID" nillable="true" type="xs:string" />
    <xs:element name="w" type="xs:int" />
    <xs:element name="x" type="xs:int" />
    <xs:element name="y" type="xs:int" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_GenericImageRegion" nillable="true"
type="tns:CT_GenericImageRegion" />
<xs:complexType name="CT_ImageRegionUse">
  <xs:sequence>
    <xs:element name="pos" nillable="true" type="tns:CT_LayoutPosition" />
    <xs:element name="reg" nillable="true" type="tns:CT_GenericImageRegion" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_ImageRegionUse" nillable="true" type="tns:CT_ImageRegionUse" />
<xs:complexType name="CT_ImageGraph">
  <xs:sequence>

```

```

        <xs:element name="base" nillable="true" type="tns:CT_ImageRegionUse" />
        <xs:element name="children" nillable="true" type="tns:ArrayOfCT_ImageRegionUse" />
    </xs:sequence>
</xs:complexType>
<xs:element name="CT_ImageGraph" nillable="true" type="tns:CT_ImageGraph" />
<xs:complexType name="ArrayOfCT_ImageRegionUse">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_ImageRegionUse"
nillable="true" type="tns:CT_ImageRegionUse" />
    </xs:sequence>
</xs:complexType>
<xs:element name="ArrayOfCT_ImageRegionUse" nillable="true"
type="tns:ArrayOfCT_ImageRegionUse" />
<xs:complexType name="CT_ImageBackground">
    <xs:sequence>
        <xs:element name="h" type="xs:int" />
        <xs:element name="imgID" nillable="true" type="xs:string" />
        <xs:element name="w" type="xs:int" />
        <xs:element name="x" type="xs:int" />
        <xs:element name="y" type="xs:int" />
    </xs:sequence>
</xs:complexType>
<xs:element name="CT_ImageBackground" nillable="true" type="tns:CT_ImageBackground" />
<xs:complexType name="CT_SolidFill">
    <xs:sequence>
        <xs:element name="clr" nillable="true" type="xs:base64Binary" />
        <xs:element name="transp" type="xs:unsignedByte" />
    </xs:sequence>
</xs:complexType>
<xs:element name="CT_SolidFill" nillable="true" type="tns:CT_SolidFill" />
<xs:complexType name="CT_BackgroundDesc">
    <xs:sequence>
        <xs:element name="Item" nillable="true" type="xs:anyType" />
    </xs:sequence>
</xs:complexType>
<xs:element name="CT_BackgroundDesc" nillable="true" type="tns:CT_BackgroundDesc" />
<xs:complexType name="CT_AnimRepeatOption">
    <xs:sequence>
        <xs:element name="count" type="xs:int" />
        <xs:element name="type" type="tns:ST_AnimRepeatType" />
    </xs:sequence>
</xs:complexType>
<xs:element name="CT_AnimRepeatOption" nillable="true" type="tns:CT_AnimRepeatOption"
/>
<xs:simpleType name="ST_AnimRepeatType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="cnt" />
        <xs:enumeration value="clk" />
        <xs:enumeration value="sld" />
    </xs:restriction>
</xs:simpleType>
<xs:element name="ST_AnimRepeatType" nillable="true" type="tns:ST_AnimRepeatType" />
<xs:complexType name="CT_ScaleBy">
    <xs:sequence>
        <xs:element name="x" type="xs:int" />
        <xs:element name="y" type="xs:int" />
    </xs:sequence>
</xs:complexType>
<xs:element name="CT_ScaleBy" nillable="true" type="tns:CT_ScaleBy" />

```

```

<xs:complexType name="CT_Anim">
  <xs:sequence>
    <xs:element name="autoRev" type="xs:boolean" />
    <xs:element name="dir" type="xs:int" />
    <xs:element name="effectId" type="xs:unsignedInt" />
    <xs:element name="effectType" type="tns:ST_EffectType" />
    <xs:element name="empImgs" nillable="true" type="tns:CT_ImageGraph" />
    <xs:element name="hideAfter" type="tns:ST_HideAfterAnimType" />
    <xs:element name="iter" nillable="true" type="tns:CT_AnimIteration" />
    <xs:element name="mp" nillable="true" type="tns:CT_MotionPath" />
    <xs:element name="repeat" nillable="true" type="tns:CT_AnimRepeatOption" />
    <xs:element name="rewind" type="xs:boolean" />
    <xs:element name="rotAng" type="xs:int" />
    <xs:element name="scaleBy" nillable="true" type="tns:CT_ScaleBy" />
    <xs:element name="sp" type="xs:unsignedInt" />
    <xs:element name="timing" nillable="true" type="tns:CT_AnimTiming" />
    <xs:element name="transp" type="xs:unsignedByte" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_Anim" nillable="true" type="tns:CT_Anim" />
<xs:simpleType name="ST_EffectType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="entr" />
    <xs:enumeration value="exit" />
    <xs:enumeration value="emph" />
    <xs:enumeration value="path" />
  </xs:restriction>
</xs:simpleType>
<xs:element name="ST_EffectType" nillable="true" type="tns:ST_EffectType" />
<xs:simpleType name="ST_HideAfterAnimType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="non" />
    <xs:enumeration value="imm" />
    <xs:enumeration value="clk" />
  </xs:restriction>
</xs:simpleType>
<xs:element name="ST_HideAfterAnimType" nillable="true" type="tns:ST_HideAfterAnimType"
/>
<xs:complexType name="CT_Transition">
  <xs:sequence>
    <xs:element name="dir" type="xs:unsignedByte" />
    <xs:element name="dur" type="xs:int" />
    <xs:element name="type" type="xs:unsignedByte" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_Transition" nillable="true" type="tns:CT_Transition" />
<xs:complexType name="CT_Shape">
  <xs:sequence>
    <xs:element name="bgSp" type="xs:boolean" />
    <xs:element name="children" nillable="true" type="tns:ArrayOfCT_ImageRegionUse" />
    <xs:element xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
name="clockTgts" nillable="true" type="q3:ArrayOfanyType" />
    <xs:element name="id" type="xs:unsignedInt" />
    <xs:element name="pos" nillable="true" type="tns:CT_LayoutPosition" />
    <xs:element name="reg" nillable="true" type="tns:CT_GenericImageRegion" />
  </xs:sequence>
</xs:complexType>
<xs:element name="CT_Shape" nillable="true" type="tns:CT_Shape" />
<xs:complexType name="CT_AnimList">

```

```

    <xs:sequence>
      <xs:element name="anim" nillable="true" type="tns:ArrayOfCT_Anim" />
      <xs:element name="id" type="xs:unsignedInt" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="CT_AnimList" nillable="true" type="tns:CT_AnimList" />
  <xs:complexType name="ArrayOfCT_Anim">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_Anim" nillable="true"
type="tns:CT_Anim" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfCT_Anim" nillable="true" type="tns:ArrayOfCT_Anim" />
  <xs:complexType name="CT_SlideShowTimings">
    <xs:sequence>
      <xs:element name="autoAdv" type="xs:boolean" />
      <xs:element name="autoAdvTime" type="xs:int" />
      <xs:element name="clickTimes" nillable="true" type="xs:string" />
      <xs:element name="manualAdv" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="CT_SlideShowTimings" nillable="true" type="tns:CT_SlideShowTimings"
/>
  <xs:complexType name="CT_Slide">
    <xs:sequence>
      <xs:element name="animLst" nillable="true" type="tns:CT_AnimList" />
      <xs:element name="animSeqs" nillable="true" type="tns:ArrayOfCT_AnimList" />
      <xs:element name="bg" nillable="true" type="tns:CT_BackgroundDesc" />
      <xs:element name="bgReg" nillable="true" type="tns:CT_GenericImageRegion" />
      <xs:element name="hidden" type="xs:boolean" />
      <xs:element name="id" type="xs:unsignedInt" />
      <xs:element name="layoutInfo" nillable="true" type="tns:CT_SlideLayoutInfo" />
      <xs:element name="notesId" nillable="true" type="xs:string" />
      <xs:element name="resCat" nillable="true" type="tns:ArrayOfCT_ResourceEntry" />
      <xs:element name="spLst" nillable="true" type="tns:ArrayOfCT_Shape" />
      <xs:element name="ssTimes" nillable="true" type="tns:CT_SlideShowTimings" />
      <xs:element name="title" nillable="true" type="xs:string" />
      <xs:element name="trans" nillable="true" type="tns:CT_Transition" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="CT_Slide" nillable="true" type="tns:CT_Slide" />
  <xs:complexType name="ArrayOfCT_AnimList">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_AnimList" nillable="true"
type="tns:CT_AnimList" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfCT_AnimList" nillable="true" type="tns:ArrayOfCT_AnimList" />
  <xs:complexType name="ArrayOfCT_ResourceEntry">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_ResourceEntry"
nillable="true" type="tns:CT_ResourceEntry" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfCT_ResourceEntry" nillable="true"
type="tns:ArrayOfCT_ResourceEntry" />
  <xs:complexType name="ArrayOfCT_Shape">
    <xs:sequence>

```



```

        <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_Shape" nillable="true"
type="tns:CT_Shape" />
    </xs:sequence>
</xs:complexType>
<xs:element name="ArrayOfCT_Shape" nillable="true" type="tns:ArrayOfCT_Shape" />
<xs:complexType name="CT_SlideEntry">
    <xs:sequence>
        <xs:element name="id" type="xs:unsignedInt" />
        <xs:element name="sldInfo" nillable="true" type="xs:string" />
        <xs:element name="title" nillable="true" type="xs:string" />
    </xs:sequence>
</xs:complexType>
<xs:element name="CT_SlideEntry" nillable="true" type="tns:CT_SlideEntry" />
<xs:complexType name="CT_EmptyType">
    <xs:sequence />
</xs:complexType>
<xs:element name="CT_EmptyType" nillable="true" type="tns:CT_EmptyType" />
<xs:complexType name="CT_IndexRange">
    <xs:sequence>
        <xs:element name="end" type="xs:unsignedInt" />
        <xs:element name="st" type="xs:unsignedInt" />
    </xs:sequence>
</xs:complexType>
<xs:element name="CT_IndexRange" nillable="true" type="tns:CT_IndexRange" />
<xs:complexType name="CT_CustomShow">
    <xs:sequence>
        <xs:element xmlns:q4="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
name="s" nillable="true" type="q4:ArrayOfunsignedInt" />
    </xs:sequence>
</xs:complexType>
<xs:element name="CT_CustomShow" nillable="true" type="tns:CT_CustomShow" />
<xs:complexType name="CT_ShowProperties">
    <xs:sequence>
        <xs:element name="Item" nillable="true" type="xs:anyType" />
    </xs:sequence>
</xs:complexType>
<xs:element name="CT_ShowProperties" nillable="true" type="tns:CT_ShowProperties" />
<xs:complexType name="CT_PresentationShowInfo">
    <xs:sequence>
        <xs:element name="h" type="xs:int" />
        <xs:element name="hasCmt" type="xs:boolean" />
        <xs:element name="rtl" type="xs:boolean" />
        <xs:element name="schmVer" type="xs:unsignedInt" />
        <xs:element name="showPr" nillable="true" type="tns:CT_ShowProperties" />
        <xs:element name="sidLst" nillable="true" type="tns:ArrayOfCT_SlideEntry" />
        <xs:element name="w" type="xs:int" />
    </xs:sequence>
</xs:complexType>
<xs:element name="CT_PresentationShowInfo" nillable="true"
type="tns:CT_PresentationShowInfo" />
    <xs:complexType name="ArrayOfCT_SlideEntry">
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_SlideEntry"
nillable="true" type="tns:CT_SlideEntry" />
        </xs:sequence>
    </xs:complexType>
    <xs:element name="ArrayOfCT_SlideEntry" nillable="true" type="tns:ArrayOfCT_SlideEntry"
/>
    <xs:complexType name="CT_SlideShowInfo">

```

```

    <xs:sequence>
      <xs:element name="errLst" nillable="true" type="tns:ArrayOfCT_Error" />
      <xs:element xmlns:q5="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
name="errMsgLst" nillable="true" type="q5:ArrayOfstring" />
      <xs:element name="schmVer" type="xs:unsignedInt" />
      <xs:element name="slid" nillable="true" type="tns:CT_Slide" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="CT_SlideShowInfo" nillable="true" type="tns:CT_SlideShowInfo" />
  <xs:complexType name="ArrayOfCT_Error">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="CT_Error" nillable="true"
type="tns:CT_Error" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfCT_Error" nillable="true" type="tns:ArrayOfCT_Error" />
  <xs:simpleType name="PowerpointView">
    <xs:restriction base="xs:string">
      <xs:enumeration value="ReadingView" />
      <xs:enumeration value="SlideShowView" />
      <xs:enumeration value="EditView" />
      <xs:enumeration value="StaticView" />
      <xs:enumeration value="OutlineView" />
      <xs:enumeration value="AttendeeView" />
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="PowerpointView" nillable="true" type="tns:PowerpointView" />
</xs:schema>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="ArrayOfstring">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfstring" nillable="true" type="tns:ArrayOfstring" />
  <xs:complexType name="ArrayOfanyType">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="anyType" nillable="true"
type="xs:anyType" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfanyType" nillable="true" type="tns:ArrayOfanyType" />
  <xs:complexType name="ArrayOfunsignedInt">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="unsignedInt"
type="xs:unsignedInt" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfunsignedInt" nillable="true" type="tns:ArrayOfunsignedInt" />
</xs:schema>
<xs:schema
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Data" elementFormDefault="qualified"
targetNamespace="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.I
nterface.Data" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="BroadcastUser">

```

```

    <xs:sequence>
      <xs:element minOccurs="0" name="SessionId" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="UserToken" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="BroadcastUser" nillable="true" type="tns:BroadcastUser" />
  <xs:complexType name="BroadcastAnimationStepData">
    <xs:sequence>
      <xs:element minOccurs="0" name="Step" type="xs:int" />
      <xs:element minOccurs="0" name="TimelineId" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="BroadcastAnimationStepData" nillable="true"
type="tns:BroadcastAnimationStepData" />
  <xs:complexType name="BroadcastData">
    <xs:sequence>
      <xs:element minOccurs="0" name="AnimationStepDataList" nillable="true"
type="tns:ArrayOfBroadcastAnimationStepData" />
      <xs:element minOccurs="0" name="HostToken" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="SequenceNumber" type="xs:int" />
      <xs:element minOccurs="0" name="SlideId" type="xs:unsignedInt" />
      <xs:element minOccurs="0" name="SlideShowState" type="tns:SlideShowState" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="BroadcastData" nillable="true" type="tns:BroadcastData" />
  <xs:complexType name="ArrayOfBroadcastAnimationStepData">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="BroadcastAnimationStepData"
nillable="true" type="tns:BroadcastAnimationStepData" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfBroadcastAnimationStepData" nillable="true"
type="tns:ArrayOfBroadcastAnimationStepData" />
  <xs:simpleType name="SlideShowState">
    <xs:restriction base="xs:string">
      <xs:enumeration value="NotStartedYet" />
      <xs:enumeration value="BlackScreen" />
      <xs:enumeration value="WhiteScreen" />
      <xs:enumeration value="Normal" />
      <xs:enumeration value="BroadcastEnded" />
      <xs:enumeration value="SlideShowEnded" />
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="SlideShowState" nillable="true" type="tns:SlideShowState" />
  <xs:complexType name="BroadcastFile">
    <xs:sequence>
      <xs:element minOccurs="0" name="FilePath" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="PresentationId" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="BroadcastFile" nillable="true" type="tns:BroadcastFile" />
  <xs:complexType name="ClientInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="Version" nillable="true" type="tns:Version" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ClientInfo" nillable="true" type="tns:ClientInfo" />
  <xs:complexType name="Version">

```

```

    <xs:sequence>
      <xs:element minOccurs="0" name="MajorNumber" type="xs:int" />
      <xs:element minOccurs="0" name="MinorNumber" type="xs:int" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="Version" nillable="true" type="tns:Version" />
  <xs:complexType name="CapabilityData">
    <xs:sequence>
      <xs:element minOccurs="0" name="Capability" type="tns:ServerCapability" />
      <xs:element minOccurs="0" name="Value" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="CapabilityData" nillable="true" type="tns:CapabilityData" />
  <xs:simpleType name="ServerCapability">
    <xs:restriction base="xs:string">
      <xs:enumeration value="SupportAudio" />
      <xs:enumeration value="SupportVideo" />
      <xs:enumeration value="SupportNotes" />
      <xs:enumeration value="SupportHandoutMaster" />
      <xs:enumeration value="FileSizeLimited" />
      <xs:enumeration value="UseWebDav" />
      <xs:enumeration value="SessionTimeout" />
      <xs:enumeration value="SessionIdleTimeout" />
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="ServerCapability" nillable="true" type="tns:ServerCapability" />
  <xs:complexType name="ServerInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="CapabilitiesList" nillable="true"
type="tns:ArrayOfCapabilityData" />
      <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="PresenterServiceUrl" nillable="true"
type="xs:string" />
      <xs:element minOccurs="0" name="Version" nillable="true" type="tns:Version" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ServerInfo" nillable="true" type="tns:ServerInfo" />
  <xs:complexType name="ArrayOfCapabilityData">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="CapabilityData"
nillable="true" type="tns:CapabilityData" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfCapabilityData" nillable="true"
type="tns:ArrayOfCapabilityData" />
  <xs:complexType name="BroadcastProtocolInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="ClientInfo" nillable="true" type="tns:ClientInfo"
/>
      <xs:element minOccurs="0" name="VersionList" nillable="true"
type="tns:ArrayOfVersion" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="BroadcastProtocolInfo" nillable="true"
type="tns:BroadcastProtocolInfo" />
  <xs:complexType name="ArrayOfVersion">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="Version" nillable="true"
type="tns:Version" />
    </xs:sequence>
  </xs:complexType>

```

```

    </xs:complexType>
    <xs:element name="ArrayOfVersion" nillable="true" type="tns:ArrayOfVersion" />
    <xs:complexType name="BroadcastHostInfo">
      <xs:sequence>
        <xs:element minOccurs="0" name="HostUrl" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="ProtocolVersion" nillable="true" type="tns:Version"
/>
      </xs:sequence>
    </xs:complexType>
    <xs:element name="BroadcastHostInfo" nillable="true" type="tns:BroadcastHostInfo" />
  </xs:schema>
</wsdl:types>
<wsdl:message name="IPptViewingService_GetPresentationInfoXml_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPresentationInfoXml" />
</wsdl:message>
<wsdl:message name="IPptViewingService_GetPresentationInfoXml_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPresentationInfoXmlResponse" />
</wsdl:message>
<wsdl:message name="IPptViewingService_GetPresentation_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPresentation" />
</wsdl:message>
<wsdl:message name="IPptViewingService_GetPresentation_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPresentationResponse" />
</wsdl:message>
<wsdl:message name="IPptViewingService_GetSlideInfo_InputMessage">
  <wsdl:part name="parameters" element="tns:GetSlideInfo" />
</wsdl:message>
<wsdl:message name="IPptViewingService_GetSlideInfo_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetSlideInfoResponse" />
</wsdl:message>
<wsdl:message name="IPptViewingService_GetSlide_InputMessage">
  <wsdl:part name="parameters" element="tns:GetSlide" />
</wsdl:message>
<wsdl:message name="IPptViewingService_GetSlide_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetSlideResponse" />
</wsdl:message>
<wsdl:message name="IPptViewingService_GetNotesHtmlServiceResult_InputMessage">
  <wsdl:part name="parameters" element="tns:GetNotesHtmlServiceResult" />
</wsdl:message>
<wsdl:message name="IPptViewingService_GetNotesHtmlServiceResult_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetNotesHtmlServiceResultResponse" />
</wsdl:message>
<wsdl:message name="IPptViewingService_BroadcastJoinSession_InputMessage">
  <wsdl:part name="parameters" element="tns:BroadcastJoinSession" />
</wsdl:message>
<wsdl:message name="IPptViewingService_BroadcastJoinSession_OutputMessage">
  <wsdl:part name="parameters" element="tns:BroadcastJoinSessionResponse" />
</wsdl:message>
<wsdl:message name="IPptViewingService_BroadcastGetData_InputMessage">
  <wsdl:part name="parameters" element="tns:BroadcastGetData" />
</wsdl:message>
<wsdl:message name="IPptViewingService_BroadcastGetData_OutputMessage">
  <wsdl:part name="parameters" element="tns:BroadcastGetDataResponse" />
</wsdl:message>
<wsdl:message name="IPptViewingService_BinaryConvert_InputMessage">
  <wsdl:part name="parameters" element="tns:BinaryConvert" />
</wsdl:message>
<wsdl:message name="IPptViewingService_BinaryConvert_OutputMessage">
  <wsdl:part name="parameters" element="tns:BinaryConvertResponse" />

```

```

</wsdl:message>
<wsdl:portType name="IPptViewingService">
  <wsdl:operation name="GetPresentationInfoXml">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetPresentationInfoXml"
message="tns:IPptViewingService_GetPresentationInfoXml_InputMessage" />
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetPresentationInfoXmlResponse"
message="tns:IPptViewingService_GetPresentationInfoXml_OutputMessage" />
    </wsdl:operation>
    <wsdl:operation name="GetPresentation">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetPresentation"
message="tns:IPptViewingService_GetPresentation_InputMessage" />
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetPresentationResponse"
message="tns:IPptViewingService_GetPresentation_OutputMessage" />
      </wsdl:operation>
      <wsdl:operation name="GetSlideInfo">
        <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetSlideInfo"
message="tns:IPptViewingService_GetSlideInfo_InputMessage" />
        <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetSlideInfoResponse"
message="tns:IPptViewingService_GetSlideInfo_OutputMessage" />
        </wsdl:operation>
        <wsdl:operation name="GetSlide">
          <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetSlide"
message="tns:IPptViewingService_GetSlide_InputMessage" />
          <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetSlideResponse"
message="tns:IPptViewingService_GetSlide_OutputMessage" />
          </wsdl:operation>
          <wsdl:operation name="GetNotesHtmlServiceResult">
            <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetNotesHtmlServiceResult"
message="tns:IPptViewingService_GetNotesHtmlServiceResult_InputMessage" />
            <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/GetNotesHtmlServiceResultResponse"
message="tns:IPptViewingService_GetNotesHtmlServiceResult_OutputMessage" />
            </wsdl:operation>
            <wsdl:operation name="BroadcastJoinSession">
              <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BroadcastJoinSession"
message="tns:IPptViewingService_BroadcastJoinSession_InputMessage" />
              <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BroadcastJoinSessionResponse"
message="tns:IPptViewingService_BroadcastJoinSession_OutputMessage" />
              </wsdl:operation>

```

```
<wsdl:operation name="BroadcastGetData">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BroadcastGetData"
message="tns:IPptViewingService_BroadcastGetData_InputMessage" />
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BroadcastGetDataResponse"
message="tns:IPptViewingService_BroadcastGetData_OutputMessage" />
  </wsdl:operation>
  <wsdl:operation name="BinaryConvert">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BinaryConvert"
message="tns:IPptViewingService_BinaryConvert_InputMessage" />
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointView
ingService/IPptViewingService/BinaryConvertResponse"
message="tns:IPptViewingService_BinaryConvert_OutputMessage" />
    </wsdl:operation>
  </wsdl:portType>
</wsdl:definitions>
```

7 Appendix B: Product Behavior

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

- Microsoft® Office Home and Business 2010
- Microsoft® Office Home and Student 2010
- Microsoft® Office Professional 2010
- Microsoft® Office Professional Plus 2010
- Microsoft® Office Standard 2010
- Microsoft® PowerPoint® 2010
- Microsoft® PowerPoint® Web App

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

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

[<1> Section 3.1:](#) This message is not supported before Microsoft Office 2010 Cumulative Update for August 2010.

[<2> Section 3.1.4:](#) This operation is not supported before Microsoft Office 2010 Cumulative Update for August 2010.

[<3> Section 3.1.4.4:](#) This operation is not supported before Microsoft Office 2010 Cumulative Update for August 2010.

8 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

9 Index

A

Abstract data model
 [client](#) 70
 [server](#) 51
[Applicability](#) 11
[ArrayOfanyType complex type](#) 15
[ArrayOfBroadcastAnimationStepData complex type](#)
 16
[ArrayOfCapabilityData complex type](#) 16
[ArrayOfCT_Anim complex type](#) 16
[ArrayOfCT_AnimList complex type](#) 17
[ArrayOfCT_Error complex type](#) 17
[ArrayOfCT_ImageRegionUse complex type](#) 17
[ArrayOfCT_ResourceEntry complex type](#) 17
[ArrayOfCT_Shape complex type](#) 18
[ArrayOfCT_SlideEntry complex type](#) 18
[ArrayOfstring complex type](#) 18
[ArrayOfunsignedInt complex type](#) 19
[ArrayOfVersion complex type](#) 19
[Attribute groups](#) 49
[Attributes](#) 48

B

[BroadcastAnimationStepData complex type](#) 19
[BroadcastData complex type](#) 20
[BroadcastFile complex type](#) 20
[BroadcastHostInfo complex type](#) 21
[BroadcastProtocolInfo complex type](#) 21

C

[Capability negotiation](#) 11
[CapabilityData complex type](#) 21
[Change tracking](#) 97
[char simple type](#) 42
Client
 [abstract data model](#) 70
 [initialization](#) 70
 [local events](#) 70
 [message processing](#) 70
 [overview](#) 50
 [sequencing rules](#) 70
 [timer events](#) 70
 [timers](#) 70
[ClientActions simple type](#) 43
[ClientInfo complex type](#) 22
[Complex types](#) 13
 [ArrayOfanyType](#) 15
 [ArrayOfBroadcastAnimationStepData](#) 16
 [ArrayOfCapabilityData](#) 16
 [ArrayOfCT_Anim](#) 16
 [ArrayOfCT_AnimList](#) 17
 [ArrayOfCT_Error](#) 17
 [ArrayOfCT_ImageRegionUse](#) 17
 [ArrayOfCT_ResourceEntry](#) 17
 [ArrayOfCT_Shape](#) 18
 [ArrayOfCT_SlideEntry](#) 18

[ArrayOfstring](#) 18
[ArrayOfunsignedInt](#) 19
[ArrayOfVersion](#) 19
[BroadcastAnimationStepData](#) 19
[BroadcastData](#) 20
[BroadcastFile](#) 20
[BroadcastHostInfo](#) 21
[BroadcastProtocolInfo](#) 21
[CapabilityData](#) 21
[ClientInfo](#) 22
[CT_Action](#) 22
[CT_Anim](#) 22
[CT_AnimIteration](#) 24
[CT_AnimList](#) 24
[CT_AnimRepeatOption](#) 24
[CT_AnimTiming](#) 25
[CT_BackgroundDesc](#) 25
[CT_CustomShow](#) 26
[CT_EmptyType](#) 26
[CT_Error](#) 26
[CT_ErrorInfo](#) 27
[CT_GenericImageRegion](#) 27
[CT_Hyperlink](#) 27
[CT_ImageBackground](#) 28
[CT_ImageGraph](#) 28
[CT_ImageRegionUse](#) 29
[CT_IndexRange](#) 29
[CT_InteractiveAnimUse](#) 30
[CT_LayoutPosition](#) 30
[CT_MotionPath](#) 30
[CT_MotionPathCurveCmd](#) 31
[CT_MotionPathLineCmd](#) 31
[CT_MotionPathMoveCmd](#) 32
[CT_PresentationShowInfo](#) 32
[CT_Rectangle](#) 33
[CT_ResourceEntry](#) 33
[CT_ScaleBy](#) 34
[CT_Shape](#) 34
[CT_ShowProperties](#) 35
[CT_Slide](#) 35
[CT_SlideEntry](#) 36
[CT_SlideLayoutInfo](#) 37
[CT_SlideShowInfo](#) 37
[CT_SlideShowTimings](#) 38
[CT_SolidFill](#) 38
[CT_Transition](#) 38
[PptViewingService.BinaryConversionResult](#) 39
[ServerInfo](#) 40
[ServiceError](#) 40
[ServiceResult](#) 41
[SlideInfoResult](#) 41
[Version](#) 41
[CT_Action complex type](#) 22
[CT_Anim complex type](#) 22
[CT_AnimIteration complex type](#) 24
[CT_AnimList complex type](#) 24
[CT_AnimRepeatOption complex type](#) 24
[CT_AnimTiming complex type](#) 25
[CT_BackgroundDesc complex type](#) 25

[CT_CustomShow_complex_type](#) 26
[CT_EmptyType_complex_type](#) 26
[CT_Error_complex_type](#) 26
[CT_ErrorInfo_complex_type](#) 27
[CT_GenericImageRegion_complex_type](#) 27
[CT_Hyperlink_complex_type](#) 27
[CT_ImageBackground_complex_type](#) 28
[CT_ImageGraph_complex_type](#) 28
[CT_ImageRegionUse_complex_type](#) 29
[CT_IndexRange_complex_type](#) 29
[CT_InteractiveAnimUse_complex_type](#) 30
[CT_LayoutPosition_complex_type](#) 30
[CT_MotionPath_complex_type](#) 30
[CT_MotionPathCurveCmd_complex_type](#) 31
[CT_MotionPathLineCmd_complex_type](#) 31
[CT_MotionPathMoveCmd_complex_type](#) 32
[CT_PresentationShowInfo_complex_type](#) 32
[CT_Rectangle_complex_type](#) 33
[CT_ResourceEntry_complex_type](#) 33
[CT_ScaleBy_complex_type](#) 34
[CT_Shape_complex_type](#) 34
[CT_ShowProperties_complex_type](#) 35
[CT_Slide_complex_type](#) 35
[CT_SlideEntry_complex_type](#) 36
[CT_SlideLayoutInfo_complex_type](#) 37
[CT_SlideShowInfo_complex_type](#) 37
[CT_SlideShowTimings_complex_type](#) 38
[CT_SolidFill_complex_type](#) 38
[CT_Transition_complex_type](#) 38

D

Data model - abstract
 [client](#) 70
 [server](#) 51
[Directory_service_schema_elements](#) 49
[duration_simple_type](#) 43

E

[Elements - directory_service_schema](#) 49

Events

[local - client](#) 70
[local - server](#) 70
[timer - client](#) 70
[timer - server](#) 70

Examples

[interaction between attendee protocol client and protocol server](#) 74
[interaction between protocol client and protocol server](#) 71
[overview](#) 71

F

[Fields - vendor-extensible](#) 11
[Full WSDL](#) 78

G

[Glossary](#) 8
[Groups](#) 48

[guid_simple_type](#) 44

I

[Implementer - security considerations](#) 77
[Index of security parameters](#) 77
[Informative references](#) 10
 Initialization
 [client](#) 70
 [server](#) 52
[Interaction between attendee protocol client and protocol server example](#) 74
[Interaction between protocol client and protocol server example](#) 71
[Introduction](#) 8

L

Local events
 [client](#) 70
 [server](#) 70

M

Message processing

[client](#) 70
[server](#) 52

Messages

[ArrayOfanyType_complex_type](#) 15
[ArrayOfBroadcastAnimationStepData_complex_type](#) 16
[ArrayOfCapabilityData_complex_type](#) 16
[ArrayOfCT_Anim_complex_type](#) 16
[ArrayOfCT_AnimList_complex_type](#) 17
[ArrayOfCT_Error_complex_type](#) 17
[ArrayOfCT_ImageRegionUse_complex_type](#) 17
[ArrayOfCT_ResourceEntry_complex_type](#) 17
[ArrayOfCT_Shape_complex_type](#) 18
[ArrayOfCT_SlideEntry_complex_type](#) 18
[ArrayOfstring_complex_type](#) 18
[ArrayOfunsignedInt_complex_type](#) 19
[ArrayOfVersion_complex_type](#) 19
[attribute groups](#) 49
[attributes](#) 48
[BroadcastAnimationStepData_complex_type](#) 19
[BroadcastData_complex_type](#) 20
[BroadcastFile_complex_type](#) 20
[BroadcastHostInfo_complex_type](#) 21
[BroadcastProtocolInfo_complex_type](#) 21
[CapabilityData_complex_type](#) 21
[char_simple_type](#) 42
[ClientActions_simple_type](#) 43
[ClientInfo_complex_type](#) 22
[complex types](#) 13
[CT_Action_complex_type](#) 22
[CT_Anim_complex_type](#) 22
[CT_AnimIteration_complex_type](#) 24
[CT_AnimList_complex_type](#) 24
[CT_AnimRepeatOption_complex_type](#) 24
[CT_AnimTiming_complex_type](#) 25
[CT_BackgroundDesc_complex_type](#) 25
[CT_CustomShow_complex_type](#) 26

[CT_EmptyType complex type](#) 26
[CT_Error complex type](#) 26
[CT_ErrorInfo complex type](#) 27
[CT_GenericImageRegion complex type](#) 27
[CT_Hyperlink complex type](#) 27
[CT_ImageBackground complex type](#) 28
[CT_ImageGraph complex type](#) 28
[CT_ImageRegionUse complex type](#) 29
[CT_IndexRange complex type](#) 29
[CT_InteractiveAnimUse complex type](#) 30
[CT_LayoutPosition complex type](#) 30
[CT_MotionPath complex type](#) 30
[CT_MotionPathCurveCmd complex type](#) 31
[CT_MotionPathLineCmd complex type](#) 31
[CT_MotionPathMoveCmd complex type](#) 32
[CT_PresentationShowInfo complex type](#) 32
[CT_Rectangle complex type](#) 33
[CT_ResourceEntry complex type](#) 33
[CT_ScaleBy complex type](#) 34
[CT_Shape complex type](#) 34
[CT_ShowProperties complex type](#) 35
[CT_Slide complex type](#) 35
[CT_SlideEntry complex type](#) 36
[CT_SlideLayoutInfo complex type](#) 37
[CT_SlideShowInfo complex type](#) 37
[CT_SlideShowTimings complex type](#) 38
[CT_SolidFill complex type](#) 38
[CT_Transition complex type](#) 38
[duration simple type](#) 43
[elements](#) 12
[enumerated](#) 12
[groups](#) 48
[guid simple type](#) 44
[namespaces](#) 12
[PowerpointView simple type](#) 44
[PptViewingService.BinaryConversionResult complex type](#) 39
[ServerCapability simple type](#) 45
[ServerInfo complex type](#) 40
[ServiceError complex type](#) 40
[ServiceErrorType simple type](#) 45
[ServiceResult complex type](#) 41
[simple types](#) 42
[SlideInfoResult complex type](#) 41
[SlideShowState simple type](#) 46
[ST_AnimRepeatType simple type](#) 46
[ST_AnimStart simple type](#) 47
[ST_EffectType simple type](#) 47
[ST_HideAfterAnimType simple type](#) 48
[syntax](#) 12
[transport](#) 12
[Version complex type](#) 41

N

[Namespaces](#) 12
[Normative references](#) 9

O

Operations
[BinaryConvert](#) 52

[BroadcastGetData](#) 54
[BroadcastJoinSession](#) 56
[BroadcastUnjoinSession](#) 58
[GetNotesHtmlServiceResult](#) 60
[GetPresentation](#) 62
[GetPresentationInfoXml](#) 64
[GetSlide](#) 66
[GetSlideInfo](#) 68
[Overview \(synopsis\)](#) 10

P

[Parameters - security index](#) 77
[PowerpointView simple type](#) 44
[PptViewingService.BinaryConversionResult complex type](#) 39
[Preconditions](#) 11
[Prerequisites](#) 11
[Product behavior](#) 96

R

References
[informative](#) 10
[normative](#) 9
[Relationship to other protocols](#) 10

S

[Schema elements - directory service](#) 49
 Security
[implementer considerations](#) 77
[parameter index](#) 77
 Sequencing rules
[client](#) 70
[server](#) 52
 Server
[abstract data model](#) 51
[BinaryConvert operation](#) 52
[BroadcastGetData operation](#) 54
[BroadcastJoinSession operation](#) 56
[BroadcastUnjoinSession operation](#) 58
[details](#) 50
[GetNotesHtmlServiceResult operation](#) 60
[GetPresentation operation](#) 62
[GetPresentationInfoXml operation](#) 64
[GetSlide operation](#) 66
[GetSlideInfo operation](#) 68
[initialization](#) 52
[local events](#) 70
[message processing](#) 52
[overview](#) 50
[sequencing rules](#) 52
[timer events](#) 70
[timers](#) 52
[ServerCapability simple type](#) 45
[ServerInfo complex type](#) 40
[ServiceError complex type](#) 40
[ServiceErrorType simple type](#) 45
[ServiceResult complex type](#) 41
[Simple types](#) 42
[char](#) 42

[ClientActions](#) 43
[duration](#) 43
[guid](#) 44
[PowerpointView](#) 44
[ServerCapability](#) 45
[ServiceErrorType](#) 45
[SlideShowState](#) 46
[ST_AnimRepeatType](#) 46
[ST_AnimStart](#) 47
[ST_EffectType](#) 47
[ST_HideAfterAnimType](#) 48
[SlideInfoResult complex type](#) 41
[SlideShowState simple type](#) 46
[ST_AnimRepeatType simple type](#) 46
[ST_AnimStart simple type](#) 47
[ST_EffectType simple type](#) 47
[ST_HideAfterAnimType simple type](#) 48
[Standards assignments](#) 11
Syntax
[messages - overview](#) 12

T

Timer events
[client](#) 70
[server](#) 70
Timers
[client](#) 70
[server](#) 52
[Tracking changes](#) 97
[Transport](#) 12
Types
[complex](#) 13
[simple](#) 42

V

[Vendor-extensible fields](#) 11
[Version complex type](#) 41
[Versioning](#) 11

W

[WSDL](#) 78