

[MS-PWEDPS]:

PowerPoint Web Editor Data Protocol

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation (“this documentation”) for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.
- **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 can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft [Open Specifications Promise](#) or the [Microsoft Community Promise](#). If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation might be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.
- **Fictitious Names.** The example companies, organizations, products, domain names, email addresses, logos, people, places, and events that are 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 as specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications documentation does 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 documents are intended for use in conjunction with publicly available standards specifications and network programming art and, as such, assume that the reader either is familiar with the aforementioned material or has immediate access to it.

Revision Summary

Date	Revision History	Revision Class	Comments
7/13/2009	0.1	Major	Initial Availability
8/28/2009	0.2	Editorial	Revised and edited the technical content
11/6/2009	0.3	Editorial	Revised and edited the technical content
2/19/2010	1.0	Major	Updated and revised the technical content
3/31/2010	1.01	Editorial	Revised and edited the technical content
4/30/2010	1.02	Editorial	Revised and edited the technical content
6/7/2010	1.03	Editorial	Revised and edited the technical content
6/29/2010	1.04	Editorial	Changed language and formatting in the technical content.
7/23/2010	1.05	Minor	Clarified the meaning of the technical content.
9/27/2010	1.05	None	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	1.06	Minor	Clarified the meaning of the technical content.
12/17/2010	1.06	None	No changes to the meaning, language, or formatting of the technical content.
3/18/2011	1.06	None	No changes to the meaning, language, or formatting of the technical content.
6/10/2011	2.0	Major	Significantly changed the technical content.
1/20/2012	3.0	Major	Significantly changed the technical content.
4/11/2012	3.0	None	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	3.0	None	No changes to the meaning, language, or formatting of the technical content.
10/8/2012	4.0	Major	Significantly changed the technical content.
2/11/2013	5.0	Major	Significantly changed the technical content.
7/30/2013	6.0	Major	Significantly changed the technical content.
11/18/2013	6.0	None	No changes to the meaning, language, or formatting of the technical content.
2/10/2014	6.0	None	No changes to the meaning, language, or formatting of the technical content.
4/30/2014	6.1	Minor	Clarified the meaning of the technical content.
7/31/2014	6.1	None	No changes to the meaning, language, or formatting of the technical content.
10/30/2014	6.1	None	No changes to the meaning, language, or formatting of the technical content.
3/16/2015	7.0	Major	Significantly changed the technical content.

Date	Revision History	Revision Class	Comments
6/23/2016	7.0	None	No changes to the meaning, language, or formatting of the technical content.
9/14/2016	7.0	None	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1	Introduction	20
1.1	Glossary	20
1.2	References	22
1.2.1	Normative References	22
1.2.2	Informative References	23
1.3	Protocol Overview (Synopsis)	23
1.4	Relationship to Other Protocols	23
1.5	Prerequisites/Preconditions	23
1.6	Applicability Statement	24
1.7	Versioning and Capability Negotiation	24
1.8	Vendor-Extensible Fields	24
1.9	Standards Assignments.....	24
2	Messages.....	25
2.1	Transport	25
2.2	Common Message Syntax	25
2.2.1	Namespaces	25
2.2.2	Messages.....	26
2.2.3	Elements	26
2.2.4	Complex Types.....	26
2.2.4.1	ArrayOfComment	28
2.2.4.2	ArrayOfContentMasterFe	29
2.2.4.3	ArrayOfEditAnimInfo	29
2.2.4.4	ArrayOfEditorInfo	29
2.2.4.5	ArrayOfEditShape	29
2.2.4.6	ArrayOfEditSlide	30
2.2.4.7	ArrayOfEditSlideChanges	30
2.2.4.8	ArrayOfEditSlideInfoFe.....	30
2.2.4.9	ArrayOfMainMasterFe	30
2.2.4.10	ArrayOfMainMasterResourcesFe	31
2.2.4.11	ArrayOfSlideUpdateInfo	31
2.2.4.12	ArrayOfStyleGalleryItem	31
2.2.4.13	ArrayOfunsignedInt	32
2.2.4.14	ArrayOfViewElement.....	32
2.2.4.15	BoundingRegion.....	32
2.2.4.16	ClippingInfo.....	33
2.2.4.17	CoauthState	33
2.2.4.18	CoauthUpdate	34
2.2.4.19	Comment	34
2.2.4.20	ContentMasterFe.....	35
2.2.4.21	ContentUpdateInfo	35
2.2.4.22	EditAnimInfo	35
2.2.4.23	EditCommandResponse	36
2.2.4.24	EditorInfo	36
2.2.4.25	EditPresentationChanges.....	37
2.2.4.26	EditPresentationInfo	38
2.2.4.27	EditPresentationResources	38
2.2.4.28	EditShape	39
2.2.4.29	EditSlide	41
2.2.4.30	EditSlideChanges	42
2.2.4.31	EditSlideInfoFe	43
2.2.4.32	EditTransitionInfo.....	44
2.2.4.33	FontSizeMapping.....	44
2.2.4.34	Html.....	45
2.2.4.35	MainMasterFe	45

2.2.4.36	MainMasterResourcesFe	46
2.2.4.37	Notes	46
2.2.4.38	Point	46
2.2.4.39	PptViewingService.PrintResult.....	47
2.2.4.40	PresetSchemeColorMapping	47
2.2.4.41	Rectangle.....	48
2.2.4.42	ServiceError	48
2.2.4.43	ServiceResult	49
2.2.4.44	SlideID	49
2.2.4.45	SlideIdList.....	49
2.2.4.46	SlideUpdateInfo	50
2.2.4.47	SmartArtInfo	50
2.2.4.48	StyleGalleryItem.....	50
2.2.4.49	Theme	51
2.2.4.50	ThemeColor.....	51
2.2.4.51	ViewElement	52
2.2.5	Simple Types	53
2.2.5.1	char	53
2.2.5.2	ClientActions	53
2.2.5.3	duration.....	54
2.2.5.4	ErrorCode	54
2.2.5.5	guid	58
2.2.5.6	ServiceErrorType	59
2.2.6	Attributes	59
2.2.7	Groups	59
2.2.8	Attribute Groups.....	59
3	Protocol Details	60
3.1	Server Details.....	60
3.1.1	Abstract Data Model.....	61
3.1.2	Timers	61
3.1.3	Initialization.....	61
3.1.4	Message Processing Events and Sequencing Rules	61
3.1.4.1	AddComment	64
3.1.4.1.1	Messages	65
3.1.4.1.1.1	IPptEdit_AddComment_InputMessage	65
3.1.4.1.1.2	IPptEdit_AddComment_OutputMessage	65
3.1.4.1.2	Elements.....	65
3.1.4.1.2.1	AddComment	65
3.1.4.1.2.2	AddCommentResponse	66
3.1.4.1.3	Complex Types	66
3.1.4.1.4	Simple Types	67
3.1.4.1.5	Attributes	67
3.1.4.1.6	Groups.....	67
3.1.4.1.7	Attribute Groups.....	67
3.1.4.2	ApplyShapeFill.....	67
3.1.4.2.1	Messages	67
3.1.4.2.1.1	IPptEdit_ApplyShapeFill_InputMessage.....	67
3.1.4.2.1.2	IPptEdit_ApplyShapeFill_OutputMessage.....	68
3.1.4.2.2	Elements.....	68
3.1.4.2.2.1	ApplyShapeFill.....	68
3.1.4.2.2.2	ApplyShapeFillResponse.....	69
3.1.4.2.3	Complex Types	69
3.1.4.2.4	Simple Types	69
3.1.4.2.5	Attributes	69
3.1.4.2.6	Groups.....	69
3.1.4.2.7	Attribute Groups.....	69
3.1.4.3	ApplyShapeOutlineColor.....	69

3.1.4.3.1	Messages	70
3.1.4.3.1.1	IPptEdit_ApplyShapeOutlineColor_InputMessage	70
3.1.4.3.1.2	IPptEdit_ApplyShapeOutlineColor_OutputMessage	70
3.1.4.3.2	Elements	70
3.1.4.3.2.1	ApplyShapeOutlineColor	70
3.1.4.3.2.2	ApplyShapeOutlineColorResponse	71
3.1.4.3.3	Complex Types	71
3.1.4.3.4	Simple Types	71
3.1.4.3.5	Attributes	72
3.1.4.3.6	Groups	72
3.1.4.3.7	Attribute Groups	72
3.1.4.4	ApplyShapeOutlineDashStyle	72
3.1.4.4.1	Messages	72
3.1.4.4.1.1	IPptEdit_ApplyShapeOutlineDashStyle_InputMessage	72
3.1.4.4.1.2	IPptEdit_ApplyShapeOutlineDashStyle_OutputMessage	73
3.1.4.4.2	Elements	73
3.1.4.4.2.1	ApplyShapeOutlineDashStyle	73
3.1.4.4.2.2	ApplyShapeOutlineDashStyleResponse	74
3.1.4.4.3	Complex Types	74
3.1.4.4.4	Simple Types	74
3.1.4.4.5	Attributes	74
3.1.4.4.6	Groups	74
3.1.4.4.7	Attribute Groups	74
3.1.4.5	ApplyShapeOutlineEndStyle	74
3.1.4.5.1	Messages	75
3.1.4.5.1.1	IPptEdit_ApplyShapeOutlineEndStyle_InputMessage	75
3.1.4.5.1.2	IPptEdit_ApplyShapeOutlineEndStyle_OutputMessage	75
3.1.4.5.2	Elements	75
3.1.4.5.2.1	ApplyShapeOutlineEndStyle	76
3.1.4.5.2.2	ApplyShapeOutlineEndStyleResponse	77
3.1.4.5.3	Complex Types	77
3.1.4.5.4	Simple Types	77
3.1.4.5.5	Attributes	77
3.1.4.5.6	Groups	77
3.1.4.5.7	Attribute Groups	77
3.1.4.6	ApplyShapeOutlineWidth	77
3.1.4.6.1	Messages	78
3.1.4.6.1.1	IPptEdit_ApplyShapeOutlineWidth_InputMessage	78
3.1.4.6.1.2	IPptEdit_ApplyShapeOutlineWidth_OutputMessage	78
3.1.4.6.2	Elements	78
3.1.4.6.2.1	ApplyShapeOutlineWidth	79
3.1.4.6.2.2	ApplyShapeOutlineWidthResponse	79
3.1.4.6.3	Complex Types	79
3.1.4.6.4	Simple Types	79
3.1.4.6.5	Attributes	80
3.1.4.6.6	Groups	80
3.1.4.6.7	Attribute Groups	80
3.1.4.7	ApplyShapeStyle	80
3.1.4.7.1	Messages	80
3.1.4.7.1.1	IPptEdit_ApplyShapeStyle_InputMessage	80
3.1.4.7.1.2	IPptEdit_ApplyShapeStyle_OutputMessage	81
3.1.4.7.2	Elements	81
3.1.4.7.2.1	ApplyShapeStyle	81
3.1.4.7.2.2	ApplyShapeStyleResponse	81
3.1.4.7.3	Complex Types	82
3.1.4.7.4	Simple Types	82
3.1.4.7.5	Attributes	82
3.1.4.7.6	Groups	82

3.1.4.7.7	Attribute Groups.....	82
3.1.4.8	ApplyTheme	82
3.1.4.8.1	Messages	82
3.1.4.8.1.1	IPptEdit_ApplyTheme_InputMessage	83
3.1.4.8.1.2	IPptEdit_ApplyTheme_OutputMessage	83
3.1.4.8.2	Elements	83
3.1.4.8.2.1	ApplyTheme	83
3.1.4.8.2.2	ApplyThemeResponse.....	84
3.1.4.8.3	Complex Types	84
3.1.4.8.4	Simple Types	84
3.1.4.8.5	Attributes	84
3.1.4.8.6	Groups.....	84
3.1.4.8.7	Attribute Groups.....	84
3.1.4.9	ArrangeShape.....	84
3.1.4.9.1	Messages	85
3.1.4.9.1.1	IPptEdit_ArrangeShape_InputMessage	85
3.1.4.9.1.2	IPptEdit_ArrangeShape_OutputMessage	85
3.1.4.9.2	Elements	85
3.1.4.9.2.1	ArrangeShape	85
3.1.4.9.2.2	ArrangeShapeResponse	86
3.1.4.9.3	Complex Types	86
3.1.4.9.4	Simple Types	86
3.1.4.9.5	Attributes	86
3.1.4.9.6	Groups.....	87
3.1.4.9.7	Attribute Groups.....	87
3.1.4.10	ChangeLayout	87
3.1.4.10.1	Messages	87
3.1.4.10.1.1	IPptEdit_ChangeLayout_InputMessage	87
3.1.4.10.1.2	IPptEdit_ChangeLayout_OutputMessage	88
3.1.4.10.2	Elements	88
3.1.4.10.2.1	ChangeLayout	88
3.1.4.10.2.2	ChangeLayoutResponse	88
3.1.4.10.3	Complex Types	89
3.1.4.10.4	Simple Types	89
3.1.4.10.5	Attributes	89
3.1.4.10.6	Groups.....	89
3.1.4.10.7	Attribute Groups.....	89
3.1.4.11	ChangePictureStyle	89
3.1.4.11.1	Messages	89
3.1.4.11.1.1	IPptEdit_ChangePictureStyle_InputMessage	90
3.1.4.11.1.2	IPptEdit_ChangePictureStyle_OutputMessage	90
3.1.4.11.2	Elements	90
3.1.4.11.2.1	ChangePictureStyle	90
3.1.4.11.2.2	ChangePictureStyleResponse.....	91
3.1.4.11.3	Complex Types	91
3.1.4.11.4	Simple Types	91
3.1.4.11.5	Attributes	91
3.1.4.11.6	Groups.....	91
3.1.4.11.7	Attribute Groups.....	91
3.1.4.12	ChangeSmartArtColor.....	91
3.1.4.12.1	Messages	92
3.1.4.12.1.1	IPptEdit_ChangeSmartArtColor_InputMessage.....	92
3.1.4.12.1.2	IPptEdit_ChangeSmartArtColor_OutputMessage.....	92
3.1.4.12.2	Elements	92
3.1.4.12.2.1	ChangeSmartArtColor.....	92
3.1.4.12.2.2	ChangeSmartArtColorResponse	93
3.1.4.12.3	Complex Types	93
3.1.4.12.4	Simple Types	93

3.1.4.12.5	Attributes	93
3.1.4.12.6	Groups.....	93
3.1.4.12.7	Attribute Groups.....	94
3.1.4.13	ChangeSmartArtLayout.....	94
3.1.4.13.1	Messages	94
3.1.4.13.1.1	IPptEdit_ChangeSmartArtLayout_InputMessage.....	94
3.1.4.13.1.2	IPptEdit_ChangeSmartArtLayout_OutputMessage	94
3.1.4.13.2	Elements	95
3.1.4.13.2.1	ChangeSmartArtLayout.....	95
3.1.4.13.2.2	ChangeSmartArtLayoutResponse	95
3.1.4.13.3	Complex Types	96
3.1.4.13.4	Simple Types	96
3.1.4.13.5	Attributes	96
3.1.4.13.6	Groups.....	96
3.1.4.13.7	Attribute Groups.....	96
3.1.4.14	ChangeSmartArtStyle	96
3.1.4.14.1	Messages	96
3.1.4.14.1.1	IPptEdit_ChangeSmartArtStyle_InputMessage	97
3.1.4.14.1.2	IPptEdit_ChangeSmartArtStyle_OutputMessage.....	97
3.1.4.14.2	Elements.....	97
3.1.4.14.2.1	ChangeSmartArtStyle.....	97
3.1.4.14.2.2	ChangeSmartArtStyleResponse.....	98
3.1.4.14.3	Complex Types	98
3.1.4.14.4	Simple Types	98
3.1.4.14.5	Attributes	98
3.1.4.14.6	Groups.....	98
3.1.4.14.7	Attribute Groups.....	98
3.1.4.15	ClearPlaceholder	98
3.1.4.15.1	Messages	99
3.1.4.15.1.1	IPptEdit_ClearPlaceholder_InputMessage	99
3.1.4.15.1.2	IPptEdit_ClearPlaceholder_OutputMessage	99
3.1.4.15.2	Elements.....	99
3.1.4.15.2.1	ClearPlaceholder	99
3.1.4.15.2.2	ClearPlaceholderResponse.....	100
3.1.4.15.3	Complex Types	100
3.1.4.15.4	Simple Types	100
3.1.4.15.5	Attributes	100
3.1.4.15.6	Groups.....	100
3.1.4.15.7	Attribute Groups.....	100
3.1.4.16	DeleteComment.....	100
3.1.4.16.1	Messages	101
3.1.4.16.1.1	IPptEdit_DeleteComment_InputMessage.....	101
3.1.4.16.1.2	IPptEdit_DeleteComment_OutputMessage.....	101
3.1.4.16.2	Elements.....	101
3.1.4.16.2.1	DeleteComment.....	102
3.1.4.16.2.2	DeleteCommentResponse	102
3.1.4.16.3	Complex Types	102
3.1.4.16.4	Simple Types	102
3.1.4.16.5	Attributes	103
3.1.4.16.6	Groups.....	103
3.1.4.16.7	Attribute Groups.....	103
3.1.4.17	DeleteSlide.....	103
3.1.4.17.1	Messages	103
3.1.4.17.1.1	IPptEdit_DeleteSlide_InputMessage	103
3.1.4.17.1.2	IPptEdit_DeleteSlide_OutputMessage	104
3.1.4.17.2	Elements.....	104
3.1.4.17.2.1	DeleteSlide	104
3.1.4.17.2.2	DeleteSlideResponse	104

3.1.4.17.3	Complex Types	104
3.1.4.17.4	Simple Types	104
3.1.4.17.5	Attributes	105
3.1.4.17.6	Groups	105
3.1.4.17.7	Attribute Groups	105
3.1.4.18	DuplicateShape	105
3.1.4.18.1	Messages	105
3.1.4.18.1.1	IPptEdit_DuplicateShape_InputMessage	105
3.1.4.18.1.2	IPptEdit_DuplicateShape_OutputMessage	106
3.1.4.18.2	Elements	106
3.1.4.18.2.1	DuplicateShape	106
3.1.4.18.2.2	DuplicateShapeResponse	107
3.1.4.18.3	Complex Types	107
3.1.4.18.4	Simple Types	107
3.1.4.18.5	Attributes	107
3.1.4.18.6	Groups	107
3.1.4.18.7	Attribute Groups	107
3.1.4.19	DuplicateSlide	107
3.1.4.19.1	Messages	108
3.1.4.19.1.1	IPptEdit_DuplicateSlide_InputMessage	108
3.1.4.19.1.2	IPptEdit_DuplicateSlide_OutputMessage	108
3.1.4.19.2	Elements	108
3.1.4.19.2.1	DuplicateSlide	108
3.1.4.19.2.2	DuplicateSlideResponse	109
3.1.4.19.3	Complex Types	109
3.1.4.19.4	Simple Types	109
3.1.4.19.5	Attributes	109
3.1.4.19.6	Groups	109
3.1.4.19.7	Attribute Groups	109
3.1.4.20	EditComment	109
3.1.4.20.1	Messages	110
3.1.4.20.1.1	IPptEdit_EditComment_InputMessage	110
3.1.4.20.1.2	IPptEdit_EditComment_OutputMessage	110
3.1.4.20.2	Elements	110
3.1.4.20.2.1	EditComment	110
3.1.4.20.2.2	EditCommentResponse	111
3.1.4.20.3	Complex Types	111
3.1.4.20.4	Simple Types	111
3.1.4.20.5	Attributes	111
3.1.4.20.6	Groups	111
3.1.4.20.7	Attribute Groups	112
3.1.4.21	FlipShape	112
3.1.4.21.1	Messages	112
3.1.4.21.1.1	IPptEdit_FlipShape_InputMessage	112
3.1.4.21.1.2	IPptEdit_FlipShape_OutputMessage	112
3.1.4.21.2	Elements	112
3.1.4.21.2.1	FlipShape	113
3.1.4.21.2.2	FlipShapeResponse	113
3.1.4.21.3	Complex Types	113
3.1.4.21.4	Simple Types	114
3.1.4.21.5	Attributes	114
3.1.4.21.6	Groups	114
3.1.4.21.7	Attribute Groups	114
3.1.4.22	GetCoauthUpdates	114
3.1.4.22.1	Messages	114
3.1.4.22.1.1	IPptEdit_GetCoauthUpdates_InputMessage	114
3.1.4.22.1.2	IPptEdit_GetCoauthUpdates_OutputMessage	115
3.1.4.22.2	Elements	115

3.1.4.22.2.1	GetCoauthUpdates	115
3.1.4.22.2.2	GetCoauthUpdatesResponse	115
3.1.4.22.3	Complex Types	116
3.1.4.22.4	Simple Types	116
3.1.4.22.5	Attributes	116
3.1.4.22.6	Groups	116
3.1.4.22.7	Attribute Groups	116
3.1.4.23	GetEditPresInfo	116
3.1.4.23.1	Messages	116
3.1.4.23.1.1	IPptEdit_GetEditPresInfo_InputMessage	117
3.1.4.23.1.2	IPptEdit_GetEditPresInfo_OutputMessage	117
3.1.4.23.2	Elements	117
3.1.4.23.2.1	GetEditPresInfo	117
3.1.4.23.2.2	GetEditPresInfoResponse	118
3.1.4.23.3	Complex Types	118
3.1.4.23.4	Simple Types	118
3.1.4.23.5	Attributes	118
3.1.4.23.6	Groups	118
3.1.4.23.7	Attribute Groups	118
3.1.4.24	GetEditPresInfoGetEditSlideById	118
3.1.4.24.1	Messages	119
3.1.4.24.1.1	IPptEdit_GetEditPresInfoGetEditSlideById_InputMessage	119
3.1.4.24.1.2	IPptEdit_GetEditPresInfoGetEditSlideById_OutputMessage	119
3.1.4.24.2	Elements	119
3.1.4.24.2.1	GetEditPresInfoGetEditSlideById	120
3.1.4.24.2.2	GetEditPresInfoGetEditSlideByIdResponse	120
3.1.4.24.3	Complex Types	121
3.1.4.24.4	Simple Types	121
3.1.4.24.5	Attributes	121
3.1.4.24.6	Groups	121
3.1.4.24.7	Attribute Groups	121
3.1.4.25	GetEditPresResources	121
3.1.4.25.1	Messages	121
3.1.4.25.1.1	IPptEdit_GetEditPresResources_InputMessage	122
3.1.4.25.1.2	IPptEdit_GetEditPresResources_OutputMessage	122
3.1.4.25.2	Elements	122
3.1.4.25.2.1	GetEditPresResources	122
3.1.4.25.2.2	GetEditPresResourcesResponse	122
3.1.4.25.3	Complex Types	123
3.1.4.25.4	Simple Types	123
3.1.4.25.5	Attributes	123
3.1.4.25.6	Groups	123
3.1.4.25.7	Attribute Groups	123
3.1.4.26	GetEditSlide	123
3.1.4.26.1	Messages	123
3.1.4.26.1.1	IPptEdit_GetEditSlide_InputMessage	124
3.1.4.26.1.2	IPptEdit_GetEditSlide_OutputMessage	124
3.1.4.26.2	Elements	124
3.1.4.26.2.1	GetEditSlide	124
3.1.4.26.2.2	GetEditSlideResponse	124
3.1.4.26.3	Complex Types	125
3.1.4.26.4	Simple Types	125
3.1.4.26.5	Attributes	125
3.1.4.26.6	Groups	125
3.1.4.26.7	Attribute Groups	125
3.1.4.27	GetPresentationId	125
3.1.4.27.1	Messages	125
3.1.4.27.1.1	IPptEdit_GetPresentationId_InputMessage	126

3.1.4.27.1.2	IPptEdit_GetPresentationId_OutputMessage.....	126
3.1.4.27.2	Elements.....	126
3.1.4.27.2.1	GetPresentationId.....	126
3.1.4.27.2.2	GetPresentationIdResponse.....	127
3.1.4.27.3	Complex Types.....	127
3.1.4.27.4	Simple Types.....	127
3.1.4.27.5	Attributes.....	127
3.1.4.27.6	Groups.....	127
3.1.4.27.7	Attribute Groups.....	127
3.1.4.28	InsertClipart.....	127
3.1.4.28.1	Messages.....	128
3.1.4.28.1.1	IPptEdit_InsertClipart_InputMessage.....	128
3.1.4.28.1.2	IPptEdit_InsertClipart_OutputMessage.....	128
3.1.4.28.2	Elements.....	128
3.1.4.28.2.1	InsertClipart.....	128
3.1.4.28.2.2	InsertClipartResponse.....	129
3.1.4.28.3	Complex Types.....	129
3.1.4.28.4	Simple Types.....	129
3.1.4.28.5	Attributes.....	129
3.1.4.28.6	Groups.....	129
3.1.4.28.7	Attribute Groups.....	129
3.1.4.29	InsertShape.....	129
3.1.4.29.1	Messages.....	130
3.1.4.29.1.1	IPptEdit_InsertShape_InputMessage.....	130
3.1.4.29.1.2	IPptEdit_InsertShape_OutputMessage.....	130
3.1.4.29.2	Elements.....	130
3.1.4.29.2.1	InsertShape.....	131
3.1.4.29.2.2	InsertShapeResponse.....	131
3.1.4.29.3	Complex Types.....	131
3.1.4.29.4	Simple Types.....	131
3.1.4.29.5	Attributes.....	132
3.1.4.29.6	Groups.....	132
3.1.4.29.7	Attribute Groups.....	132
3.1.4.30	InsertSlide.....	132
3.1.4.30.1	Messages.....	132
3.1.4.30.1.1	IPptEdit_InsertSlide_InputMessage.....	132
3.1.4.30.1.2	IPptEdit_InsertSlide_OutputMessage.....	133
3.1.4.30.2	Elements.....	133
3.1.4.30.2.1	InsertSlide.....	133
3.1.4.30.2.2	InsertSlideResponse.....	133
3.1.4.30.3	Complex Types.....	134
3.1.4.30.4	Simple Types.....	134
3.1.4.30.5	Attributes.....	134
3.1.4.30.6	Groups.....	134
3.1.4.30.7	Attribute Groups.....	134
3.1.4.31	InsertSmartArt.....	134
3.1.4.31.1	Messages.....	134
3.1.4.31.1.1	IPptEdit_InsertSmartArt_InputMessage.....	135
3.1.4.31.1.2	IPptEdit_InsertSmartArt_OutputMessage.....	135
3.1.4.31.2	Elements.....	135
3.1.4.31.2.1	InsertSmartArt.....	135
3.1.4.31.2.2	InsertSmartArtResponse.....	136
3.1.4.31.3	Complex Types.....	136
3.1.4.31.4	Simple Types.....	136
3.1.4.31.5	Attributes.....	136
3.1.4.31.6	Groups.....	136
3.1.4.31.7	Attribute Groups.....	136
3.1.4.32	LogULS.....	136

3.1.4.32.1	Messages	137
3.1.4.32.1.1	IPptEdit_LogULS_InputMessage.....	137
3.1.4.32.1.2	IPptEdit_LogULS_OutputMessage.....	137
3.1.4.32.2	Elements.....	137
3.1.4.32.2.1	LogULS.....	137
3.1.4.32.2.2	LogULSResponse	138
3.1.4.32.3	Complex Types	138
3.1.4.32.4	Simple Types	138
3.1.4.32.5	Attributes	138
3.1.4.32.6	Groups.....	138
3.1.4.32.7	Attribute Groups.....	138
3.1.4.33	MoveComment	138
3.1.4.33.1	Messages	139
3.1.4.33.1.1	IPptEdit_MoveComment_InputMessage	139
3.1.4.33.1.2	IPptEdit_MoveComment_OutputMessage	139
3.1.4.33.2	Elements.....	139
3.1.4.33.2.1	MoveComment	139
3.1.4.33.2.2	MoveCommentResponse	140
3.1.4.33.3	Complex Types	140
3.1.4.33.4	Simple Types	140
3.1.4.33.5	Attributes	140
3.1.4.33.6	Groups.....	141
3.1.4.33.7	Attribute Groups.....	141
3.1.4.34	MoveShape	141
3.1.4.34.1	Messages	141
3.1.4.34.1.1	IPptEdit_MoveShape_InputMessage	141
3.1.4.34.1.2	IPptEdit_MoveShape_OutputMessage	141
3.1.4.34.2	Elements.....	142
3.1.4.34.2.1	MoveShape	142
3.1.4.34.2.2	MoveShapeResponse.....	142
3.1.4.34.3	Complex Types	142
3.1.4.34.4	Simple Types	143
3.1.4.34.5	Attributes	143
3.1.4.34.6	Groups.....	143
3.1.4.34.7	Attribute Groups.....	143
3.1.4.35	MoveSlide	143
3.1.4.35.1	Messages	143
3.1.4.35.1.1	IPptEdit_MoveSlide_InputMessage	143
3.1.4.35.1.2	IPptEdit_MoveSlide_OutputMessage	144
3.1.4.35.2	Elements.....	144
3.1.4.35.2.1	MoveSlide	144
3.1.4.35.2.2	MoveSlideResponse.....	144
3.1.4.35.3	Complex Types	145
3.1.4.35.4	Simple Types	145
3.1.4.35.5	Attributes	145
3.1.4.35.6	Groups.....	145
3.1.4.35.7	Attribute Groups.....	145
3.1.4.36	PastePicture	145
3.1.4.36.1	Messages	145
3.1.4.36.1.1	IPptEdit_PastePicture_InputMessage	146
3.1.4.36.1.2	IPptEdit_PastePicture_OutputMessage	146
3.1.4.36.2	Elements.....	146
3.1.4.36.2.1	PastePicture	146
3.1.4.36.2.2	PastePictureResponse.....	147
3.1.4.36.3	Complex Types	147
3.1.4.36.4	Simple Types	147
3.1.4.36.5	Attributes	147
3.1.4.36.6	Groups.....	147

3.1.4.36.7	Attribute Groups.....	147
3.1.4.37	Print.....	147
3.1.4.37.1	Messages	148
3.1.4.37.1.1	IPptEdit_Print_InputMessage.....	148
3.1.4.37.1.2	IPptEdit_Print_OutputMessage.....	148
3.1.4.37.2	Elements	148
3.1.4.37.2.1	Print.....	148
3.1.4.37.2.2	PrintResponse	149
3.1.4.37.3	Complex Types	149
3.1.4.37.4	Simple Types	149
3.1.4.37.5	Attributes	149
3.1.4.37.6	Groups.....	149
3.1.4.37.7	Attribute Groups.....	149
3.1.4.38	Redo	149
3.1.4.38.1	Messages	150
3.1.4.38.1.1	IPptEdit_Redo_InputMessage	150
3.1.4.38.1.2	IPptEdit_Redo_OutputMessage	150
3.1.4.38.2	Elements	150
3.1.4.38.2.1	Redo	150
3.1.4.38.2.2	RedoResponse	151
3.1.4.38.3	Complex Types	151
3.1.4.38.4	Simple Types	151
3.1.4.38.5	Attributes	151
3.1.4.38.6	Groups.....	151
3.1.4.38.7	Attribute Groups.....	151
3.1.4.39	RemoveShapeFill	151
3.1.4.39.1	Messages	152
3.1.4.39.1.1	IPptEdit_RemoveShapeFill_InputMessage	152
3.1.4.39.1.2	IPptEdit_RemoveShapeFill_OutputMessage	152
3.1.4.39.2	Elements	152
3.1.4.39.2.1	RemoveShapeFill	152
3.1.4.39.2.2	RemoveShapeFillResponse	153
3.1.4.39.3	Complex Types	153
3.1.4.39.4	Simple Types	153
3.1.4.39.5	Attributes	153
3.1.4.39.6	Groups.....	153
3.1.4.39.7	Attribute Groups.....	153
3.1.4.40	RemoveShapeOutline	154
3.1.4.40.1	Messages	154
3.1.4.40.1.1	IPptEdit_RemoveShapeOutline_InputMessage	154
3.1.4.40.1.2	IPptEdit_RemoveShapeOutline_OutputMessage	154
3.1.4.40.2	Elements	154
3.1.4.40.2.1	RemoveShapeOutline	155
3.1.4.40.2.2	RemoveShapeOutlineResponse	155
3.1.4.40.3	Complex Types	155
3.1.4.40.4	Simple Types	155
3.1.4.40.5	Attributes	156
3.1.4.40.6	Groups.....	156
3.1.4.40.7	Attribute Groups.....	156
3.1.4.41	ReorderAnimation	156
3.1.4.41.1	Messages	156
3.1.4.41.1.1	IPptEdit_ReorderAnimation_InputMessage	156
3.1.4.41.1.2	IPptEdit_ReorderAnimation_OutputMessage	157
3.1.4.41.2	Elements	157
3.1.4.41.2.1	ReorderAnimation	157
3.1.4.41.2.2	ReorderAnimationResponse.....	157
3.1.4.41.3	Complex Types	158
3.1.4.41.4	Simple Types	158

3.1.4.41.5	Attributes	158
3.1.4.41.6	Groups.....	158
3.1.4.41.7	Attribute Groups.....	158
3.1.4.42	ReplaceNotes	158
3.1.4.42.1	Messages	158
3.1.4.42.1.1	IPptEdit_ReplaceNotes_InputMessage	159
3.1.4.42.1.2	IPptEdit_ReplaceNotes_OutputMessage	159
3.1.4.42.2	Elements.....	159
3.1.4.42.2.1	ReplaceNotes	159
3.1.4.42.2.2	ReplaceNotesResponse	160
3.1.4.42.3	Complex Types	160
3.1.4.42.4	Simple Types	160
3.1.4.42.5	Attributes	160
3.1.4.42.6	Groups.....	160
3.1.4.42.7	Attribute Groups.....	160
3.1.4.43	ReplaceText	160
3.1.4.43.1	Messages	161
3.1.4.43.1.1	IPptEdit_ReplaceText_InputMessage	161
3.1.4.43.1.2	IPptEdit_ReplaceText_OutputMessage	161
3.1.4.43.2	Elements.....	161
3.1.4.43.2.1	ReplaceText	161
3.1.4.43.2.2	ReplaceTextResponse	162
3.1.4.43.3	Complex Types	162
3.1.4.43.4	Simple Types	162
3.1.4.43.5	Attributes	162
3.1.4.43.6	Groups.....	162
3.1.4.43.7	Attribute Groups.....	162
3.1.4.44	ResetPicture	162
3.1.4.44.1	Messages	163
3.1.4.44.1.1	IPptEdit_ResetPicture_InputMessage	163
3.1.4.44.1.2	IPptEdit_ResetPicture_OutputMessage.....	163
3.1.4.44.2	Elements.....	163
3.1.4.44.2.1	ResetPicture	164
3.1.4.44.2.2	ResetPictureResponse.....	164
3.1.4.44.3	Complex Types	164
3.1.4.44.4	Simple Types	164
3.1.4.44.5	Attributes	164
3.1.4.44.6	Groups.....	164
3.1.4.44.7	Attribute Groups.....	164
3.1.4.45	ResetSmartArt.....	165
3.1.4.45.1	Messages	165
3.1.4.45.1.1	IPptEdit_ResetSmartArt_InputMessage.....	165
3.1.4.45.1.2	IPptEdit_ResetSmartArt_OutputMessage.....	165
3.1.4.45.2	Elements.....	165
3.1.4.45.2.1	ResetSmartArt.....	166
3.1.4.45.2.2	ResetSmartArtResponse	166
3.1.4.45.3	Complex Types	166
3.1.4.45.4	Simple Types	166
3.1.4.45.5	Attributes	167
3.1.4.45.6	Groups.....	167
3.1.4.45.7	Attribute Groups.....	167
3.1.4.46	ResizeShape.....	167
3.1.4.46.1	Messages	167
3.1.4.46.1.1	IPptEdit_ResizeShape_InputMessage.....	167
3.1.4.46.1.2	IPptEdit_ResizeShape_OutputMessage	168
3.1.4.46.2	Elements.....	168
3.1.4.46.2.1	ResizeShape.....	168
3.1.4.46.2.2	ResizeShapeResponse	169

3.1.4.46.3	Complex Types	169
3.1.4.46.4	Simple Types	169
3.1.4.46.5	Attributes	169
3.1.4.46.6	Groups	169
3.1.4.46.7	Attribute Groups	169
3.1.4.47	ReverseSmartArt	169
3.1.4.47.1	Messages	170
3.1.4.47.1.1	IPptEdit_ReverseSmartArt_InputMessage	170
3.1.4.47.1.2	IPptEdit_ReverseSmartArt_OutputMessage	170
3.1.4.47.2	Elements	170
3.1.4.47.2.1	ReverseSmartArt	170
3.1.4.47.2.2	ReverseSmartArtResponse	171
3.1.4.47.3	Complex Types	171
3.1.4.47.4	Simple Types	171
3.1.4.47.5	Attributes	171
3.1.4.47.6	Groups	171
3.1.4.47.7	Attribute Groups	171
3.1.4.48	RotateShape	171
3.1.4.48.1	Messages	172
3.1.4.48.1.1	IPptEdit_RotateShape_InputMessage	172
3.1.4.48.1.2	IPptEdit_RotateShape_OutputMessage	172
3.1.4.48.2	Elements	172
3.1.4.48.2.1	RotateShape	172
3.1.4.48.2.2	RotateShapeResponse	173
3.1.4.48.3	Complex Types	173
3.1.4.48.4	Simple Types	173
3.1.4.48.5	Attributes	173
3.1.4.48.6	Groups	173
3.1.4.48.7	Attribute Groups	173
3.1.4.49	SaveAndClose	174
3.1.4.49.1	Messages	174
3.1.4.49.1.1	IPptEdit_SaveAndClose_InputMessage	174
3.1.4.49.1.2	IPptEdit_SaveAndClose_OutputMessage	174
3.1.4.49.2	Elements	174
3.1.4.49.2.1	SaveAndClose	175
3.1.4.49.2.2	SaveAndCloseResponse	175
3.1.4.49.3	Complex Types	175
3.1.4.49.4	Simple Types	175
3.1.4.49.5	Attributes	175
3.1.4.49.6	Groups	176
3.1.4.49.7	Attribute Groups	176
3.1.4.50	SetAnimation	176
3.1.4.50.1	Messages	176
3.1.4.50.1.1	IPptEdit_SetAnimation_InputMessage	176
3.1.4.50.1.2	IPptEdit_SetAnimation_OutputMessage	177
3.1.4.50.2	Elements	177
3.1.4.50.2.1	SetAnimation	177
3.1.4.50.2.2	SetAnimationResponse	178
3.1.4.50.3	Complex Types	178
3.1.4.50.4	Simple Types	178
3.1.4.50.5	Attributes	178
3.1.4.50.6	Groups	179
3.1.4.50.7	Attribute Groups	179
3.1.4.51	SetShapeAlignment	179
3.1.4.51.1	Messages	179
3.1.4.51.1.1	IPptEdit_SetShapeAlignment_InputMessage	179
3.1.4.51.1.2	IPptEdit_SetShapeAlignment_OutputMessage	180
3.1.4.51.2	Elements	180

3.1.4.51.2.1	SetShapeAlignment.....	180
3.1.4.51.2.2	SetShapeAlignmentResponse	180
3.1.4.51.3	Complex Types	181
3.1.4.51.4	Simple Types	181
3.1.4.51.5	Attributes	181
3.1.4.51.6	Groups.....	181
3.1.4.51.7	Attribute Groups.....	181
3.1.4.52	SetShapeBold	181
3.1.4.52.1	Messages	181
3.1.4.52.1.1	IPptEdit_SetShapeBold_InputMessage.....	182
3.1.4.52.1.2	IPptEdit_SetShapeBold_OutputMessage.....	182
3.1.4.52.2	Elements	182
3.1.4.52.2.1	SetShapeBold.....	182
3.1.4.52.2.2	SetShapeBoldResponse	183
3.1.4.52.3	Complex Types	183
3.1.4.52.4	Simple Types	183
3.1.4.52.5	Attributes	183
3.1.4.52.6	Groups.....	183
3.1.4.52.7	Attribute Groups.....	183
3.1.4.53	SetShapeBullet	183
3.1.4.53.1	Messages	184
3.1.4.53.1.1	IPptEdit_SetShapeBullet_InputMessage.....	184
3.1.4.53.1.2	IPptEdit_SetShapeBullet_OutputMessage.....	184
3.1.4.53.2	Elements	184
3.1.4.53.2.1	SetShapeBullet.....	184
3.1.4.53.2.2	SetShapeBulletResponse.....	185
3.1.4.53.3	Complex Types	185
3.1.4.53.4	Simple Types	185
3.1.4.53.5	Attributes	186
3.1.4.53.6	Groups.....	186
3.1.4.53.7	Attribute Groups.....	186
3.1.4.54	SetShapeFontColor.....	186
3.1.4.54.1	Messages	186
3.1.4.54.1.1	IPptEdit_SetShapeFontColor_InputMessage	186
3.1.4.54.1.2	IPptEdit_SetShapeFontColor_OutputMessage	187
3.1.4.54.2	Elements.....	187
3.1.4.54.2.1	SetShapeFontColor	187
3.1.4.54.2.2	SetShapeFontColorResponse	187
3.1.4.54.3	Complex Types	188
3.1.4.54.4	Simple Types	188
3.1.4.54.5	Attributes	188
3.1.4.54.6	Groups.....	188
3.1.4.54.7	Attribute Groups.....	188
3.1.4.55	SetShapeFontName.....	188
3.1.4.55.1	Messages	188
3.1.4.55.1.1	IPptEdit_SetShapeFontName_InputMessage.....	189
3.1.4.55.1.2	IPptEdit_SetShapeFontName_OutputMessage	189
3.1.4.55.2	Elements.....	189
3.1.4.55.2.1	SetShapeFontName.....	189
3.1.4.55.2.2	SetShapeFontNameResponse	190
3.1.4.55.3	Complex Types	190
3.1.4.55.4	Simple Types	190
3.1.4.55.5	Attributes	190
3.1.4.55.6	Groups.....	190
3.1.4.55.7	Attribute Groups.....	190
3.1.4.56	SetShapeFontSize	190
3.1.4.56.1	Messages	191
3.1.4.56.1.1	IPptEdit_SetShapeFontSize_InputMessage	191

3.1.4.56.1.2	IPptEdit_SetShapeFontSize_OutputMessage.....	191
3.1.4.56.2	Elements.....	191
3.1.4.56.2.1	SetShapeFontSize.....	191
3.1.4.56.2.2	SetShapeFontSizeResponse.....	192
3.1.4.56.3	Complex Types.....	192
3.1.4.56.4	Simple Types.....	192
3.1.4.56.5	Attributes.....	192
3.1.4.56.6	Groups.....	192
3.1.4.56.7	Attribute Groups.....	192
3.1.4.57	SetShapeItalic.....	193
3.1.4.57.1	Messages.....	193
3.1.4.57.1.1	IPptEdit_SetShapeItalic_InputMessage.....	193
3.1.4.57.1.2	IPptEdit_SetShapeItalic_OutputMessage.....	193
3.1.4.57.2	Elements.....	193
3.1.4.57.2.1	SetShapeItalic.....	194
3.1.4.57.2.2	SetShapeItalicResponse.....	194
3.1.4.57.3	Complex Types.....	194
3.1.4.57.4	Simple Types.....	194
3.1.4.57.5	Attributes.....	195
3.1.4.57.6	Groups.....	195
3.1.4.57.7	Attribute Groups.....	195
3.1.4.58	SetShapeTextDirection.....	195
3.1.4.58.1	Messages.....	195
3.1.4.58.1.1	IPptEdit_SetShapeTextDirection_InputMessage.....	195
3.1.4.58.1.2	IPptEdit_SetShapeTextDirection_OutputMessage.....	196
3.1.4.58.2	Elements.....	196
3.1.4.58.2.1	SetShapeTextDirection.....	196
3.1.4.58.2.2	SetShapeTextDirectionResponse.....	196
3.1.4.58.3	Complex Types.....	197
3.1.4.58.4	Simple Types.....	197
3.1.4.58.5	Attributes.....	197
3.1.4.58.6	Groups.....	197
3.1.4.58.7	Attribute Groups.....	197
3.1.4.59	SetShapeUnderline.....	197
3.1.4.59.1	Messages.....	198
3.1.4.59.1.1	IPptEdit_SetShapeUnderline_InputMessage.....	198
3.1.4.59.1.2	IPptEdit_SetShapeUnderline_OutputMessage.....	198
3.1.4.59.2	Elements.....	198
3.1.4.59.2.1	SetShapeUnderline.....	198
3.1.4.59.2.2	SetShapeUnderlineResponse.....	199
3.1.4.59.3	Complex Types.....	199
3.1.4.59.4	Simple Types.....	199
3.1.4.59.5	Attributes.....	199
3.1.4.59.6	Groups.....	199
3.1.4.59.7	Attribute Groups.....	199
3.1.4.60	SetTransition.....	199
3.1.4.60.1	Messages.....	200
3.1.4.60.1.1	IPptEdit_SetTransition_InputMessage.....	200
3.1.4.60.1.2	IPptEdit_SetTransition_OutputMessage.....	200
3.1.4.60.2	Elements.....	200
3.1.4.60.2.1	SetTransition.....	201
3.1.4.60.2.2	SetTransitionResponse.....	202
3.1.4.60.3	Complex Types.....	202
3.1.4.60.4	Simple Types.....	202
3.1.4.60.5	Attributes.....	202
3.1.4.60.6	Groups.....	202
3.1.4.60.7	Attribute Groups.....	202
3.1.4.61	SetTransitionApplyAll.....	202

3.1.4.61.1	Messages	203
3.1.4.61.1.1	IPptEdit_SetTransitionApplyAll_InputMessage	203
3.1.4.61.1.2	IPptEdit_SetTransitionApplyAll_OutputMessage	203
3.1.4.61.2	Elements	203
3.1.4.61.2.1	SetTransitionApplyAll	203
3.1.4.61.2.2	SetTransitionApplyAllResponse	204
3.1.4.61.3	Complex Types	204
3.1.4.61.4	Simple Types	204
3.1.4.61.5	Attributes	204
3.1.4.61.6	Groups	204
3.1.4.61.7	Attribute Groups	204
3.1.4.62	SetWordArt	204
3.1.4.62.1	Messages	205
3.1.4.62.1.1	IPptEdit_SetWordArt_InputMessage	205
3.1.4.62.1.2	IPptEdit_SetWordArt_OutputMessage	205
3.1.4.62.2	Elements	205
3.1.4.62.2.1	SetWordArt	205
3.1.4.62.2.2	SetWordArtResponse	206
3.1.4.62.3	Complex Types	206
3.1.4.62.4	Simple Types	206
3.1.4.62.5	Attributes	206
3.1.4.62.6	Groups	206
3.1.4.62.7	Attribute Groups	206
3.1.4.63	ShapeFormatPainting	207
3.1.4.63.1	Messages	207
3.1.4.63.1.1	IPptEdit_ShapeFormatPainting_InputMessage	207
3.1.4.63.1.2	IPptEdit_ShapeFormatPainting_OutputMessage	207
3.1.4.63.2	Elements	207
3.1.4.63.2.1	ShapeFormatPainting	208
3.1.4.63.2.2	ShapeFormatPaintingResponse	208
3.1.4.63.3	Complex Types	209
3.1.4.63.4	Simple Types	209
3.1.4.63.5	Attributes	209
3.1.4.63.6	Groups	209
3.1.4.63.7	Attribute Groups	209
3.1.4.64	ShowHideSlide	209
3.1.4.64.1	Messages	209
3.1.4.64.1.1	IPptEdit_ShowHideSlide_InputMessage	210
3.1.4.64.1.2	IPptEdit_ShowHideSlide_OutputMessage	210
3.1.4.64.2	Elements	210
3.1.4.64.2.1	ShowHideSlide	210
3.1.4.64.2.2	ShowHideSlideResponse	210
3.1.4.64.3	Complex Types	211
3.1.4.64.4	Simple Types	211
3.1.4.64.5	Attributes	211
3.1.4.64.6	Groups	211
3.1.4.64.7	Attribute Groups	211
3.1.4.65	Undo	211
3.1.4.65.1	Messages	211
3.1.4.65.1.1	IPptEdit_Undo_InputMessage	212
3.1.4.65.1.2	IPptEdit_Undo_OutputMessage	212
3.1.4.65.2	Elements	212
3.1.4.65.2.1	Undo	212
3.1.4.65.2.2	UndoResponse	212
3.1.4.65.3	Complex Types	213
3.1.4.65.4	Simple Types	213
3.1.4.65.5	Attributes	213
3.1.4.65.6	Groups	213

3.1.4.65.7	Attribute Groups.....	213
3.1.4.66	UngroupShape.....	213
3.1.4.66.1	Messages	213
3.1.4.66.1.1	IPptEdit_UngroupShape_InputMessage.....	214
3.1.4.66.1.2	IPptEdit_UngroupShape_OutputMessage	214
3.1.4.66.2	Elements	214
3.1.4.66.2.1	UngroupShape.....	214
3.1.4.66.2.2	UngroupShapeResponse	215
3.1.4.66.3	Complex Types	215
3.1.4.66.4	Simple Types	215
3.1.4.66.5	Attributes	215
3.1.4.66.6	Groups.....	215
3.1.4.66.7	Attribute Groups.....	215
3.1.4.67	UpdateCanary.....	215
3.1.4.67.1	Messages	216
3.1.4.67.1.1	IPptEdit_UpdateCanary_InputMessage	216
3.1.4.67.1.2	IPptEdit_UpdateCanary_OutputMessage	216
3.1.4.67.2	Elements	216
3.1.4.67.2.1	UpdateCanary	216
3.1.4.67.2.2	UpdateCanaryResponse	217
3.1.4.67.3	Complex Types	217
3.1.4.67.4	Simple Types	217
3.1.4.67.5	Attributes	217
3.1.4.67.6	Groups.....	217
3.1.4.67.7	Attribute Groups.....	217
3.1.5	Timer Events.....	217
3.1.6	Other Local Events.....	217
4	Protocol Examples	218
4.1	Sample Protocol Interaction.....	218
5	Security.....	228
5.1	Security Considerations for Implementers	228
5.2	Index of Security Parameters	228
6	Appendix A: Full WSDL	229
7	Appendix B: Full XML Schema.....	256
7.1	http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared.Schema	256
7.2	http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.Interface.Schema	257
7.3	http://schemas.datacontract.org/2004/07/p.Schema	258
7.4	http://schemas.microsoft.com/2003/10/Serialization/Arrays.Schema	266
7.5	http://schemas.microsoft.com/2003/10/Serialization/Schema	266
7.6	http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/Schema	267
8	Appendix C: Product Behavior	286
9	Change Tracking.....	291
10	Index.....	292

1 Introduction

The PowerPoint Web Editor Data Protocol enables a protocol client to obtain information about and modify slide collection content on a protocol server.

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

1.1 Glossary

This document uses the following terms:

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

ActiveX control: A reusable software control, such as a check box or button, that uses ActiveX technology and provides options to users or runs macros or scripts that automate a task. See also ActiveX object.

build identifier: An integer that identifies a build (1).

canary: A token that stores data about an application or user session. Typically it is used by web services to help authenticate an application or user.

comment: An annotation that is associated with a cell, text, or other object to provide context-specific information or reviewer feedback.

cookie: A small data file that is stored on a user's computer and carries state information between participating protocol servers and protocol clients.

cross-site request forgery: A type of security vulnerability that enables malicious users to send unauthorized commands to a website or application by assuming the identity of an authorized user and running script under that assumed identity. Also referred to as one-click attack and session riding.

cross-site scripting: A type of security vulnerability that enables malicious users to insert client-side script into webpages and to run that script when those pages are viewed by other users. The script might then gain access to user-specific data, such as cookies, cached objects, and application settings. Also referred to as XSS.

Hypertext Markup Language (HTML): An application of the Standard Generalized Markup Language (SGML) that uses tags to mark elements in a document, as described in [\[HTML\]](#).

ink: A process of entering text in handwritten form. Instead of converting handwritten text to typed text, ink is converted to an object and displayed exactly as it was written.

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

main master slide: A slide that defines the formatting and content that can be used by presentation slides. If a slide uses formatting and content from a main master slide, it is referred to as following a main master slide.

OLE object: An object that supports the Object Linking and Embedding (OLE) protocol.

placeholder: A character or symbol that is used in place of an actual value, text, or object. The actual value that the placeholder represents is unknown or unavailable at the current time, or is not displayed for security reasons.

presentation: A collection of slides that are intended to be viewed by an audience.

presentation slide: A slide that contains the content that can be displayed during a slide show. A presentation slide can derive formatting and content from a main master slide or a title master slide.

right-to-left: A reading and display order that is optimized for right-to-left languages.

shape: A collection of qualifiers, such as names, and quantifiers, such as coordinates, that is used to represent a geometric object. A shape can be contained in a document, file structure, run-time structure, or other medium.

slide: A frame that contains text, shapes, pictures, or other content. A slide is a digital equivalent to a traditional film slide.

slide layout: An organizational scheme, such as Title Only or Comparison, for content on a presentation slide.

SmartArt: A type of graphical element, such as a diagram or image, that automatically positions and combines images with text to produce a well-formatted and well-designed graphic.

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

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

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

SOAP message: An XML document consisting of a mandatory SOAP envelope, an optional SOAP header, and a mandatory **SOAP body**. See [\[SOAP1.2-1/2007\]](#) section 5 for more information.

theme: A set of unified design elements, such as colors, fonts, graphics, and styles, that define the appearance of a website, document, or data visualization.

title master slide: A slide that defines the formatting and content that can be used by presentation slides that have a title slide layout. If a slide uses formatting and content from a title master slide, it is referred to as following a title master slide.

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

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

website: A group of related webpages that is hosted by a server on the World Wide Web or an intranet. Each website has its own entry points, metadata, administration settings, and workflows. Also referred to as site.

WSDL message: An abstract, typed definition of the data that is communicated during a **WSDL operation** [\[WSDL\]](#). Also, an element that describes the data being exchanged between web service providers and clients.

WSDL operation: A single action or function of a web service. The execution of a WSDL operation typically requires the exchange of messages between the service requestor and the service provider.

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

XML namespace prefix: An abbreviated form of an **XML namespace**, as described in [XML].

XML schema: A description of a type of XML document that is typically expressed in terms of constraints on the structure and content of documents of that type, in addition to the basic syntax constraints that are imposed by XML itself. An XML schema provides a view of a document type at a relatively high level of abstraction.

z-order: The rendering order of an object on a z axis.

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

1.2 References

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

1.2.1 Normative References

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

[ISO/IEC29500-1:2011] ISO/IEC, "Information Technology -- Document description and processing languages -- Office Open XML File Formats -- Part 1: Fundamentals and Markup Language Reference", ISO/IEC 29500-1:2011, 2011, http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=59575

[MS-LCID] Microsoft Corporation, "[Windows Language Code Identifier \(LCID\) Reference](#)".

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

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

[RFC4627] Crockford, D., "The application/json Media Type for JavaScript Object Notation (JSON)", RFC 4627, July 2006, <http://www.rfc-editor.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>

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

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

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

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

1.2.2 Informative References

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

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

1.3 Protocol Overview (Synopsis)

This protocol enables a protocol client to send a request to modify **presentation** content to the protocol server and then receive from the protocol server information about the result of the modification.

To facilitate this, the protocol allows the protocol client to send a request to the protocol server and then receive from the protocol server information about the existence of **presentation slides**, **main master slides**, and **slide layouts**. The protocol client can also send a request to the protocol server and then receive from the protocol server images of **slide** content.

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:

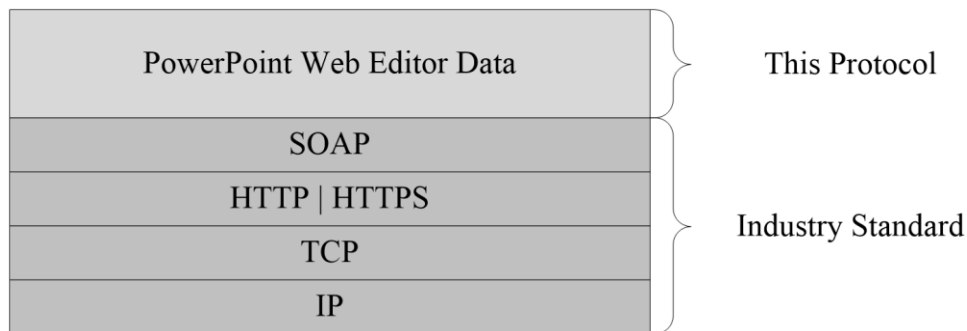


Figure 1: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against a set of **Web sites** that is identified by a set of **URLs** that are known by protocol clients.

The protocol server endpoint to which the client sends requests to modify presentation content is formed by appending "/p/ppt/edit.svc" to the URL of the Web site, for example <http://www.contoso.com/p/ppt/edit.svc>.

The protocol server endpoint to which the client sends requests to modify image content is formed by appending "/p/pptInsertPicture.ashx" to the URL of the Web site, for example <http://www.contoso.com/p/pptInsertPicture.ashx>.

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

1.6 Applicability Statement

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

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following area:

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

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

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

2.1 Transport

Protocol servers **MUST** support SOAP over HTTP. Protocol servers **SHOULD** additionally support SOAP over HTTPS for securing communication with clients. Protocol servers **SHOULD** support JSON [\[RFC4627\]](#) over HTTP and HTTPS. <1>

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 WSDL 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
q3	http://schemas.microsoft.com/2003/10/Serialization/Arrays	
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
tns	http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared	
tns1	http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.Interface	
tns2	http://schemas.datacontract.org/2004/07/p	
tns3	http://schemas.microsoft.com/2003/10/Serialization/	
tns4	http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/	
tns5	http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/Imports	
wsaw	http://www.w3.org/2006/05/addressing/wsdl	
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]

Pref ix	Namespace URI	Reference
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEM A1] [XMLSCHEM A2]

2.2.2 Messages

This specification does not define any common **WSDL message** definitions.

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
ArrayOfComment	The ArrayOfComment complex type <2> specifies a list of Comment elements.
ArrayOfContentMasterFe	The ArrayOfContentMasterFe complex type specifies a list of ContentMasterFe elements.
ArrayOfEditAnimInfo	The ArrayOfEditAnimInfo complex type <3> specifies a list of EditAnimInfo elements.
ArrayOfEditorInfo	The ArrayOfEditorInfo complex type <4> specifies a list of EditorInfo elements.
ArrayOfEditShape	The ArrayOfEditShape complex type specifies a list of EditShape elements.
ArrayOfEditSlide	The ArrayOfEditSlide complex type specifies a list of EditSlide elements.
ArrayOfEditSlideChanges	The ArrayOfEditSlideChanges complex type specifies a list of EditSlideChanges elements.
ArrayOfEditSlideInfoFe	The ArrayOfEditSlideInfoFe complex type specifies a list of EditSlideInfoFe elements.
ArrayOfMainMasterFe	The ArrayOfMainMasterFe complex type specifies a list of MainMasterFe elements.
ArrayOfMainMasterResourcesFe	The ArrayOfMainMasterResourcesFe specifies a list of MainMasterResourcesFe elements.
ArrayOfSlideUpdateInfo	The ArrayOfSlideUpdateInfo specifies a list of SlideUpdateInfo elements.
ArrayOfStyleGalleryItem	The ArrayOfStyleGalleryItem complex type <5> specifies a list of StyleGalleryItem elements.

Complex type	Description
ArrayOfunsignedInt	The ArrayOfunsignedInt complex type specifies a list of unsignedInt elements ([XMLSCHEMA2] section 3.3.22).
ArrayOfViewElement	The ArrayOfViewElement complex type specifies a list of ViewElement elements.
BoundingRegion	The BoundingRegion complex type <6> specifies the rotated bounds of a shape .
ClippingInfo	The ClippingInfo complex type <7> specifies a region within an image.
CoauthState	The CoauthState complex type <8> specifies the co-authoring state of the presentation.
CoauthUpdate	The CoauthUpdate complex type <9> specifies the co-authoring state of the presentation.
Comment	The Comment complex type <10> specifies a single comment , and its replies.
ContentMasterFe	The ContentMasterFe complex type specifies a slide layout.
ContentUpdateInfo	Reserved. MUST be ignored.
EditAnimInfo	The EditAnimInfo complex type <11> specifies a shape animation.
EditCommandResponse	The EditCommandResponse complex type specifies the response of a modification to a presentation.
EditorInfo	The EditorInfo complex type <12> specifies an editor of the presentation.
EditPresentationChanges	The EditPresentationChanges complex type specifies the content of the presentation that is modified as a result of a request to the protocol server.
EditPresentationInfo	The EditPresentationInfo complex type specifies information about the content of a specified presentation.
EditPresentationResources	Reserved. MUST be ignored.
EditShape	The EditShape complex type specifies a shape.
EditSlide	The EditSlide complex type specifies a presentation slide.
EditSlideChanges	The EditSlideChanges complex type specifies the content of the presentation slide. The content of this element is modified as a result of a request to the protocol server.
EditSlideInfoFe	The EditSlideInfoFe complex type specifies information about a presentation slide.
EditTransitionInfo	The EditTransitionInfo complex type <13> specifies a slide transition.
FontSizeMapping	Reserved; this element MUST be ignored.
Html	The Html complex type specifies HTML content.
MainMasterFe	The MainMasterFe complex type specifies a main master slide.
MainMasterResourcesFe	Reserved. MUST be ignored.
Notes	The Notes complex type specifies the notes of a presentation slide.

Complex type	Description
Point	The Point complex type specifies a two dimensional point.
PptViewingService.PrintResult	The PptViewingService.PrintResult complex type <14> specifies the result of a print operation.
PresetSchemeColorMapping	The PresetSchemeColorMapping complex type <15> specifies a set of colors in the theme .
Rectangle	The Rectangle complex type specifies a rectangle.
ServiceError	The ServiceError complex type specifies error information. This information is returned by the protocol server to a protocol client.
ServiceResult	The ServiceResult complex type 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.
SlideID	A complex type that SHOULD <16> identify a presentation slide.
SlideIdList	The SlideIdList complex type specifies a list of identifiers for presentation slides.
SlideUpdateInfo	Reserved. MUST be ignored.
SmartArtInfo	The SmartArtInfo complex type specifies information about a SmartArt diagram.
StyleGalleryItem	The StyleGalleryItem complex type <17> specifies a shape style.
Theme	The Theme complex type specifies information about a theme.
ThemeColor	The ThemeColor complex type specifies a color using the red, green, blue RGB color model. Red, green, and blue are each expressed as an unsigned byte from 0 for no intensity to 255 for full intensity.
ViewElement	The ViewElement complex type specifies a visual representation of presentation slide content.

2.2.4.1 ArrayOfComment

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

The **ArrayOfComment** complex type [<18>](#) specifies a list of **Comment** elements.

```
<xs:complexType name="ArrayOfComment" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="Comment" nillable="true"
      type="tns2:Comment"/>
  </xs:sequence>
</xs:complexType>
```

Comment: Each element MUST specify a **Comment** complex type (section [2.2.4.19](#)) instance.

2.2.4.2 ArrayOfContentMasterFe

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

The **ArrayOfContentMasterFe** complex type specifies a list of **ContentMasterFe** elements.

```
<xs:complexType name="ArrayOfContentMasterFe" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="ContentMasterFe" nillable="true"
type="tns2:ContentMasterFe"/>
  </xs:sequence>
</xs:complexType>
```

ContentMasterFe: Each element MUST specify a **ContentMasterFe** complex type (section [2.2.4.20](#)) instance.

2.2.4.3 ArrayOfEditAnimInfo

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

The **ArrayOfEditAnimInfo** complex type [<19>](#) specifies a list of **EditAnimInfo** elements.

```
<xs:complexType name="ArrayOfEditAnimInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="EditAnimInfo" nillable="true"
type="tns2:EditAnimInfo"/>
  </xs:sequence>
</xs:complexType>
```

EditAnimInfo: Each element MUST specify an **EditAnimInfo** complex type (section [2.2.4.22](#)) instance.

2.2.4.4 ArrayOfEditorInfo

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

The **ArrayOfEditorInfo** complex type [<20>](#) specifies a list of **EditorInfo** elements.

```
<xs:complexType name="ArrayOfEditorInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="EditorInfo" nillable="true"
type="tns2:EditorInfo"/>
  </xs:sequence>
</xs:complexType>
```

EditorInfo: Each element MUST specify an **EditorInfo** complex type (section [2.2.4.24](#)) instance.

2.2.4.5 ArrayOfEditShape

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

The **ArrayOfEditShape** complex type specifies a list of **EditShape** elements.

```
<xs:complexType name="ArrayOfEditShape" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="EditShape" nillable="true"
type="tns2:EditShape"/>
  </xs:sequence>
</xs:complexType>
```

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

EditShape: Each element MUST specify an **EditShape** complex type (section [2.2.4.28](#)) instance.

2.2.4.6 ArrayOfEditSlide

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

The **ArrayOfEditSlide** complex type specifies a list of **EditSlide** elements.

```
<xs:complexType name="ArrayOfEditSlide" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="EditSlide" nillable="true"
      type="tns2:EditSlide"/>
  </xs:sequence>
</xs:complexType>
```

EditSlide: Each element MUST specify an **EditSlide** complex type (section [2.2.4.29](#)) instance.

2.2.4.7 ArrayOfEditSlideChanges

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

The **ArrayOfEditSlideChanges** complex type specifies a list of **EditSlideChanges** elements.

```
<xs:complexType name="ArrayOfEditSlideChanges" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="EditSlideChanges" nillable="true"
      type="tns2:EditSlideChanges"/>
  </xs:sequence>
</xs:complexType>
```

EditSlideChanges: Each element MUST specify an **EditSlideChanges** complex type (section [2.2.4.30](#)) instance.

2.2.4.8 ArrayOfEditSlideInfoFe

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

The **ArrayOfEditSlideInfoFe** complex type specifies a list of **EditSlideInfoFe** elements.

```
<xs:complexType name="ArrayOfEditSlideInfoFe" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="EditSlideInfoFe" nillable="true"
      type="tns2:EditSlideInfoFe"/>
  </xs:sequence>
</xs:complexType>
```

EditSlideInfoFe: Each element MUST specify an **EditSlideInfoFe** complex type (section [2.2.4.31](#)) instance.

2.2.4.9 ArrayOfMainMasterFe

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

The **ArrayOfMainMasterFe** complex type specifies a list of **MainMasterFe** elements.

```
<xs:complexType name="ArrayOfMainMasterFe" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="MainMasterFe" nillable="true"
type="tns2:MainMasterFe"/>
  </xs:sequence>
</xs:complexType>
```

MainMasterFe: Each element MUST specify a **MainMasterFe** complex type (section [2.2.4.35](#)) instance.

2.2.4.10 ArrayOfMainMasterResourcesFe

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

The **ArrayOfMainMasterResourcesFe** specifies a list of **MainMasterResourcesFe** elements.

```
<xs:complexType name="ArrayOfMainMasterResourcesFe"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="MainMasterResourcesFe"
nillable="true" type="tns2:MainMasterResourcesFe"/>
  </xs:sequence>
</xs:complexType>
```

MainMasterResourcesFe: Each element MUST specify a **MainMasterResourcesFe** complex type (section [2.2.4.36](#)) instance.

2.2.4.11 ArrayOfSlideUpdateInfo

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

The **ArrayOfSlideUpdateInfo** specifies a list of **SlideUpdateInfo** elements.

```
<xs:complexType name="ArrayOfSlideUpdateInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="SlideUpdateInfo" nillable="true"
type="tns2:SlideUpdateInfo"/>
  </xs:sequence>
</xs:complexType>
```

SlideUpdateInfo: Each element MUST specify a **SlideUpdateInfo** complex type (section [2.2.4.46](#)) instance.

2.2.4.12 ArrayOfStyleGalleryItem

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

The **ArrayOfStyleGalleryItem** complex type [<21>](#) specifies a list of **StyleGalleryItem** elements.

```
<xs:complexType name="ArrayOfStyleGalleryItem" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="StyleGalleryItem" nillable="true"
type="tns2:StyleGalleryItem"/>
  </xs:sequence>
```

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

StyleGalleryItem: Each element MUST specify a **StyleGalleryItem** complex type (section [2.2.4.48](#)) instance.

2.2.4.13 ArrayOfunsignedInt

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

The **ArrayOfunsignedInt** complex type specifies a list of **unsignedInt** elements ([\[XMLSCHEMA2\]](#) section 3.3.22).

```
<xs:complexType name="ArrayOfunsignedInt" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="unsignedInt"
      type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
```

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

2.2.4.14 ArrayOfViewElement

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

The **ArrayOfViewElement** complex type specifies a list of **ViewElement** elements.

```
<xs:complexType name="ArrayOfViewElement" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="ViewElement" nillable="true"
      type="tns2:ViewElement"/>
  </xs:sequence>
</xs:complexType>
```

ViewElement: Each element MUST specify a **View Element** complex type (section [2.2.4.51](#)) instance.

2.2.4.15 BoundingRegion

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

The **BoundingRegion** complex type [<22>](#) specifies the rotated bounds of a shape.

```
<xs:complexType name="BoundingRegion" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="BottomLeft" nillable="true" type="tns2:Point"/>
    <xs:element minOccurs="0" name="BottomRight" nillable="true" type="tns2:Point"/>
    <xs:element minOccurs="0" name="TopLeft" nillable="true" type="tns2:Point"/>
    <xs:element minOccurs="0" name="TopRight" nillable="true" type="tns2:Point"/>
  </xs:sequence>
</xs:complexType>
```

BottomLeft: A **Point** complex element (section [2.2.4.38](#)) that specifies the bottom left point of the shape. This element MUST be present.

BottomRight: A **Point** element that specifies the bottom right point of the shape. This element **MUST** be present.

TopLeft: A **Point** element that specifies the top left point of the shape. This element **MUST** be present.

TopRight: A **Point** element that specifies the top right point of the shape. This element **MUST** be present.

2.2.4.16 ClippingInfo

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

The **ClippingInfo** complex type [<23>](#) specifies a region within an image.

```
<xs:complexType name="ClippingInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Bounds" nillable="true" type="tns2:Rectangle"/>
    <xs:element minOccurs="0" name="TotalHeight" type="xs:int"/>
    <xs:element minOccurs="0" name="TotalWidth" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

Bounds: A **Rectangle** complex element (section [2.2.4.41](#)) that specifies the bounds of the region. This element **MUST** be present.

TotalHeight: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the total height of the image.

TotalWidth: An **int** that specifies the total width of the image.

2.2.4.17 CoauthState

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

The **CoauthState** complex type [<24>](#) specifies the co-authoring state of the presentation.

```
<xs:complexType name="CoauthState" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="cs" type="xs:int"/>
    <xs:element minOccurs="0" name="et" nillable="true" type="tns2:ArrayOfEditorInfo"/>
    <xs:element minOccurs="0" name="lsave" type="xs:dateTime"/>
    <xs:element minOccurs="0" name="lsmce" type="xs:dateTime"/>
    <xs:element minOccurs="0" name="lsum" type="xs:dateTime"/>
    <xs:element minOccurs="0" name="meid" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="sui" nillable="true" type="tns2:ContentUpdateInfo"/>
  </xs:sequence>
</xs:complexType>
```

cs: Reserved. **MUST** be ignored.

et: An **ArrayOfEditorInfo** complex element (section [2.2.4.4](#)) that specifies the list of editors that are currently co-authoring the presentation.

lsave: Reserved. **MUST** be ignored.

lsmce: Reserved. **MUST** be ignored.

lsum: Reserved. **MUST** be ignored.

meid: Reserved. MUST be ignored.

sui: Reserved. MUST be ignored.

2.2.4.18 CoauthUpdate

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

The **CoauthUpdate** complex type [<25>](#) specifies the co-authoring state of the presentation.

```
<xs:complexType name="CoauthUpdate" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="State" nillable="true" type="tns2:CoauthState"/>
    <xs:element minOccurs="0" name="Token" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

State: A **CoauthState** complex element (section [2.2.4.17](#)) that specifies the co-authoring state of the presentation.

Token: Reserved. Must be ignored.

2.2.4.19 Comment

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

The **Comment** complex type [<26>](#) specifies a single comment, and its replies.

```
<xs:complexType name="Comment" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Children" nillable="true" type="tns2:ArrayOfComment"/>
    <xs:element minOccurs="0" name="authorId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="authorName" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="dt" type="xs:dateTime"/>
    <xs:element minOccurs="0" name="idx" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="pos" nillable="true" type="tns2:Point"/>
    <xs:element minOccurs="0" name="text" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

Children: An **ArrayOfComment** complex type (section [2.2.4.1](#)) that contains the replies to this comment.

authorId: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the identifier of the author of the comment. This element MUST be present.

authorName: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the name of the author of the comment.

dt: A **dateTime** ([\[XMLSCHEMA2\]](#) section 3.2.7) that specifies the time in UTC in which the comment was created.

idx: An **unsignedInt** that specifies the identifier of the comment within its author's comments. This element MUST be present.

pos: A **Point** complex type (section [2.2.4.38](#)) that specifies the position of the comment on the slide.

text: A **string** that specifies the contents of the comment.

2.2.4.20 ContentMasterFe

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

The **ContentMasterFe** complex type specifies a slide layout.

```
<xs:complexType name="ContentMasterFe" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Id" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="SlideLayoutType" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

Id: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the identifier for the slide layout. This element **MUST** be present.

Name: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the name of the slide layout.

SlideLayoutType: A **string** that specifies the layout type as specified by **ST_SlideLayoutType** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.15). This element **MUST** be present.

2.2.4.21 ContentUpdateInfo

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

Reserved. **MUST** be ignored.

```
<xs:complexType name="ContentUpdateInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="leid" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="lplu" type="xs:dateTime"/>
    <xs:element minOccurs="0" name="redo" type="xs:boolean"/>
    <xs:element minOccurs="0" name="sui" nillable="true" type="tns2:ArrayOfSlideUpdateInfo"/>
    <xs:element minOccurs="0" name="undo" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
```

leid: Reserved. **MUST** be ignored.

lplu: Reserved. **MUST** be ignored.

redo: Reserved. **MUST** be ignored.

sui: Reserved. **MUST** be ignored.

undo: Reserved. **MUST** be ignored.

2.2.4.22 EditAnimInfo

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

The **EditAnimInfo** complex type [<27>](#) specifies a shape animation.

```
<xs:complexType name="EditAnimInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
      minOccurs="0" name="BuildIds" nillable="true" type="q3:ArrayOfunsignedInt"/>
    <xs:element minOccurs="0" name="Direction" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
```

```

<xs:element minOccurs="0" name="EffectId" type="xs:unsignedInt"/>
<xs:element minOccurs="0" name="HasExisting" type="xs:boolean"/>
<xs:element minOccurs="0" name="HasTrigger" type="xs:boolean"/>
<xs:element minOccurs="0" name="SequenceId" type="xs:unsignedInt"/>
<xs:element minOccurs="0" name="ShapeId" type="xs:unsignedInt"/>
</xs:sequence>
</xs:complexType>

```

BuildIds: An **ArrayOfunsignedInt** complex element (section [2.2.4.13](#)) that specifies the list of **build identifiers**. This element MUST be present.

Direction: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the direction of animation. This element MAY be present.

EffectId: An **unsignedInt** that specifies the type of animation. This element MUST be present.

HasExisting: Reserved. MUST be ignored.

HasTrigger: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that specifies whether the shape has one or more animations using a triggered sequence. This element MUST be present.

SequenceId: An **unsignedInt** that specifies the type of animation. This element MUST be present.

ShapeId: An **unsignedInt** that specifies the identifier of the shape. This element MUST be present.

2.2.4.23 EditCommandResponse

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

The **EditCommandResponse** complex type specifies the response of a modification to a presentation.

```

<xs:complexType name="EditCommandResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="PresChanges" nillable="true"
type="tns2:EditPresentationChanges"/>
  </xs:sequence>
</xs:complexType>

```

PresChanges: An **EditPresentationChanges** complex type (section [2.2.4.25](#)) that specifies the content of the presentation that is modified.

2.2.4.24 EditorInfo

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

The **EditorInfo** complex type [<28>](#) specifies an editor of the presentation.

```

<xs:complexType name="EditorInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="EmailAddress" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="ID" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="SIPAddress" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Timeout" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="UserName" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

```

EmailAddress: Reserved. MUST be ignored.

ID: A string ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the editor.

Name: Reserved. MUST be ignored.

SIPAddress: Reserved. MUST be ignored.

TimeOut: Reserved. MUST be ignored.

UserName: Reserved. MUST be ignored.

2.2.4.25 EditPresentationChanges

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

The **EditPresentationChanges** complex type specifies the content of the presentation that is modified as a result of a request to the protocol server.

```
<xs:complexType name="EditPresentationChanges" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="AddedSlideInfos" nillable="true"
type="tns2:ArrayOfEditSlideInfoFe"/>
    <xs:element minOccurs="0" name="AddedSlides" nillable="true"
type="tns2:ArrayOfEditSlide"/>
    <xs:element minOccurs="0" name="InvalidateAllThumbs" type="xs:boolean"/>
    <xs:element minOccurs="0" name="MainMasterIds" nillable="true" type="tns2:SlideIdList"/>
    <xs:element minOccurs="0" name="ModifiedSlides" nillable="true"
type="tns2:ArrayOfEditSlideChanges"/>
    <xs:element minOccurs="0" name="Redo" type="xs:boolean"/>
    <xs:element minOccurs="0" name="SlideOrder" nillable="true" type="tns2:SlideIdList"/>
    <xs:element minOccurs="0" name="Undo" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
```

AddedSlideInfos: An **ArrayOfEditSlideInfoFe** complex element (section [2.2.4.8](#)) that specifies information about presentation slides added to the presentation. This element MUST be present.

AddedSlides: An **ArrayOfEditSlide** complex element (section [2.2.4.5](#)) that specifies presentation slides added to the presentation. This element MUST be present.

InvalidateAllThumbs: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that SHOULD [<29>](#) specify whether all thumbnails are invalid. This element MUST be present.

MainMasterIds: A **SlideIdList** complex element (section [2.2.4.45](#)) that specifies main master slides that are referenced by presentation slides.

ModifiedSlides: An **ArrayOfEditSlideChanges** complex element (section [2.2.4.7](#)) that specifies slide content that is modified. This element MUST be present.

Redo: A **boolean** that specifies whether a **Redo** operation (section [3.1.4.38](#)) is possible. This element MUST be present.

SlideOrder: A **SlideIdList** complex element (section [2.2.4.45](#)) that specifies the order of the presentation slides.

Undo: A **boolean** that specifies whether an **Undo** operation (section [3.1.4.65](#)) is possible. This element MUST be present.

2.2.4.26 EditPresentationInfo

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

The **EditPresentationInfo** complex type specifies information about the content of a specified presentation.

```
<xs:complexType name="EditPresentationInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="CompatProblems" type="xs:boolean"/>
    <xs:element minOccurs="0" name="EditSlideInfos" nillable="true"
type="tns2:ArrayOfEditSlideInfoFe"/>
    <xs:element minOccurs="0" name="FirstSlideIndex" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="MainMasters" nillable="true"
type="tns2:ArrayOfMainMasterFe"/>
    <xs:element minOccurs="0" name="Redo" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Rtl" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Size" nillable="true" type="tns2:Point"/>
    <xs:element minOccurs="0" name="Slide" nillable="true" type="tns2:EditSlide"/>
    <xs:element minOccurs="0" name="SlideOrder" nillable="true" type="tns2:SlideIdList"/>
    <xs:element minOccurs="0" name="Undo" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
```

CompatProblems: A **boolean** ([XMLSCHEMA2] section 3.2.2) that SHOULD [<30>](#) specify whether compatibility issues can occur if the presentation is edited. This element MUST be present.

EditSlideInfos: An **ArrayOfEditSlideInfoFe** complex element (section [2.2.4.8](#)) that specifies information about presentation slides. This element MUST be present.

FirstSlideIndex: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the index of the first presentation slide. This element MUST be present.

MainMasters: An **ArrayOfMainMasterFe** complex element (section [2.2.4.9](#)) that specify information about main master slides. This element MUST be present.

Redo: A **boolean** that specifies whether a **Redo** operation (section [3.1.4.38](#)) is possible. This element MUST be present.

Rtl: A **boolean** that specifies whether the presentation content is **right-to-left**. This element MUST be present.

Size: A **Point** complex element (section [2.2.4.38](#)) that specifies the dimensions of the presentation slides. This element MUST be present.

Slide: Reserved. MUST be ignored.

SlideOrder: A **SlideIdList** complex element (section [2.2.4.45](#)) that specifies the order of the presentation slides. This element MUST be present.

Undo: A **boolean** that specifies whether an **Undo** operation (section [3.1.4.65](#)) is possible. This element MUST be present.

2.2.4.27 EditPresentationResources

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

Reserved. MUST be ignored.

```
<xs:complexType name="EditPresentationResources" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
```

```

    <xs:element minOccurs="0" name="MainMasterResources" nillable="true"
type="tns2:ArrayOfMainMasterResourcesFe"/>
  </xs:sequence>
</xs:complexType>

```

MainMasterResources: Reserved. MUST be ignored.

2.2.4.28 EditShape

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

The **EditShape** complex type specifies a shape.

```

<xs:complexType name="EditShape" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Alt" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="EditBounds" nillable="true" type="tns2:Rectangle"/>
    <xs:element minOccurs="0" name="EditHtml" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="FFlippedHorizontally" type="xs:boolean"/>
    <xs:element minOccurs="0" name="FFlippedVertically" type="xs:boolean"/>
    <xs:element minOccurs="0" name="FontName" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="FontSize" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="FontSizeMapping" nillable="true"
type="tns2:FontSizeMapping"/>
    <xs:element minOccurs="0" name="HasEffectAndNoFill" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsActiveX" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsBold" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsChart" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsConnected" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsGroup" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsInk" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsItalic" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsLine" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsNonNativeVideo" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsOleObject" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsPicture" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsPlaceholder" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsPrompt" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsSmartArt" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsStraightLine" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsTable" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsTextbox" type="xs:boolean"/>
    <xs:element minOccurs="0" name="IsUnderline" type="xs:boolean"/>
    <xs:element minOccurs="0" name="PlaceholderType" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="RotatedBoundingRegion" nillable="true"
type="tns2:BoundingRegion"/>
    <xs:element minOccurs="0" name="Rotation" type="xs:int"/>
    <xs:element minOccurs="0" name="ShapeId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="ShapeViewElemId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="SmartArtInfo" nillable="true" type="tns2:SmartArtInfo"/>
    <xs:element minOccurs="0" name="TextAlignmentType" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="TextBulletType" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="TextDirectionType" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="TextViewElemId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="X" type="xs:double"/>
    <xs:element minOccurs="0" name="Y" type="xs:double"/>
    <xs:element minOccurs="0" name="Z" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>

```

Alt: Reserved. MUST be ignored.

EditBounds: A **Rectangle** complex type (section [2.2.4.41](#)) that specifies the bounds of the shape. This element MUST be present.

EditHtml: A **string** ([XMLSCHEMA2] section 3.2 1) that specifies the text content of the shape as HTML. This element **MUST** be present.

FflippedHorizontally: A **boolean** ([XMLSCHEMA2] section 3.2.2) that specifies whether the shape is currently flipped horizontally. This element **MUST** be present.

FflippedVertically: A **boolean** that specifies whether the shape is currently flipped vertically. This element **MUST** be present.

FontName: A **string** ([XMLSCHEMA2] section 3.2 1) that specifies the font name of the text for the shape.

FontSize: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the font size of the text for the shape.

FontSizeMapping: Reserved. **MUST** be ignored.

HasEffectAndNoFill: Reserved. **MUST** be ignored.

IsActiveX: A **boolean** that specifies whether the shape is an **ActiveX control**. This element **MUST** be present.

IsBold: A **boolean** that specifies whether the shape text is bold.

IsChart: A **boolean** that specifies whether the shape is a chart. This element **MUST** be present.

IsConnected: A **boolean** that specifies whether the shape is a connector that is attached to one or more shapes. This element **MUST** be present.

IsGroup: A **boolean** that specifies whether the shape is a group of shapes. This element **MUST** be present.

IsInk: A **boolean** that specifies whether the shape is an **ink** drawing. This element **MUST** be present.

IsItalic: A **boolean** that specifies whether the shape text is italicized.

IsLine: A **boolean** that specifies whether the shape is a line. This element **MUST** be present.

IsNonNativeVideo: Reserved. **MUST** be ignored.

IsOleObject: A **boolean** that specifies whether the shape is an **OLE object**. This element **MUST** be present.

IsPicture: A **boolean** that specifies whether the shape is a picture. This element **MUST** be present.

IsPlaceholder: A **boolean** that specifies whether the shape is a **placeholder**. This element **MUST** be present.

IsPrompt: A **boolean** that specifies whether the shape contains text. This element **MUST** be present.

IsSmartArt: A **boolean** that specifies whether the shape is a SmartArt diagram. This element **MUST** be present.

IsStraightLine: A **boolean** that specifies whether the shape is a straight line. This element **MUST** be present.

IsTable: A **boolean** that specifies whether the shape is a table. This element **MUST** be present.

IsTextbox: A **boolean** that specifies whether the shape is a text box. This element **MUST** be present.

IsUnderline: A **boolean** that specifies whether the shape text is underlined.

PlaceholderType: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the type of placeholder. This element **MUST** be present.

RotatedBoundingRegion: A **BoundingRegion** complex element (section [2.2.4.15](#)) that specifies the orientation of the shape. This element **MUST** be present.

Rotation: An **int** ([XMLSCHEMA2] section 3.3.17) that specifies the rotation of the shape around its center in degrees.

ShapeId: An **unsignedInt** that specifies the identifier of the shape. This element **MUST** be present.

ShapeViewElemId: An **unsignedInt** that specifies the identifier of the **ViewElement** complex element (section [2.2.4.51](#)) for the shape. This element **MUST** be present.

SmartArtInfo: A **SmartArtInfo** complex element (section [2.2.4.47](#)) that specifies information about the SmartArt diagram. This element **MUST** be present if and only if **IsSmartArt** is true.

TextAlignmentType: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the type of text alignment.

TextBulletType: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the type of text bullet.

TextDirectionType: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the direction of text.

TextViewElemId: An **unsignedInt** that specifies the identifier of the **ViewElement** for the text content of the shape.

X: A **double** ([XMLSCHEMA2] section 3.3.5) that specifies the x coordinate of the shape.

Y: A **double** that specifies the y coordinate of the shape.

Z: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the **z-order** of the shape.

2.2.4.29 EditSlide

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

The **EditSlide** complex type specifies a presentation slide.

```
<xs:complexType name="EditSlide" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Comments" nillable="true" type="tns2:ArrayOfComment"/>
    <xs:element minOccurs="0" name="ContentMasterId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Created" type="xs:dateTime"/>
    <xs:element minOccurs="0" name="EditAnims" nillable="true"
type="tns2:ArrayOfEditAnimInfo"/>
    <xs:element minOccurs="0" name="EditShapes" nillable="true"
type="tns2:ArrayOfEditShape"/>
    <xs:element minOccurs="0" name="EditTransition" nillable="true"
type="tns2:EditTransitionInfo"/>
    <xs:element minOccurs="0" name="Hidden" type="xs:boolean"/>
    <xs:element minOccurs="0" name="ID" type="tns2:SlideID"/>
    <xs:element minOccurs="0" name="IsTitleContentMaster" type="xs:boolean"/>
    <xs:element minOccurs="0" name="MainMasterId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Notes" nillable="true" type="tns2:Notes"/>
    <xs:element minOccurs="0" name="SlideId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Theme" nillable="true" type="tns2:Theme"/>
    <xs:element minOccurs="0" name="ThumbnailViewElement" nillable="true"
type="tns2:ViewElement"/>
    <xs:element minOccurs="0" name="ViewElems" nillable="true"
type="tns2:ArrayOfViewElement"/>
  </xs:sequence>
</xs:complexType>
```

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

Comments: An **ArrayOfComment** complex element (section [2.2.4.1](#)) that specifies comments for the presentation slide.

ContentMasterId: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the identifier of the **ContentMasterFe** complex element (section [2.2.4.20](#)) for the presentation slide. This element MUST be present.

Created: Reserved. MUST be ignored.

EditAnims: An **ArrayOfEditAnimInfo** complex element (section [2.2.4.3](#)) that specifies the list of animations on the slide.

EditShapes: An **ArrayOfEditShape** complex element (section [2.2.4.5](#)) that specifies the shapes of the presentation slide. This element MUST be present.

EditTransition: An **EditTransitionInfo** complex element (section [2.2.4.32](#)) that specifies the slide transition.

Hidden: A **boolean** ([XMLSCHEMA2] section 3.2.2) that specifies whether the presentation slide is hidden. This element MUST be present.

ID: A **SlideID** (section [2.2.4.44](#)) that specifies the identifier of the presentation slide.

IsTitleContentMaster: A **boolean** that specifies whether the slide layout specified by **ContentMasterId** is a **title master slide**. This element MUST be present.

MainMasterId: An **unsignedInt** that specifies the identifier of the main master slide for this presentation slide. This element MUST be present.

Notes: A **Notes** complex element (section [2.2.4.37](#)) that specifies the presentation notes for the presentation slide. This element MUST be present.

SlideId: An **unsignedInt** that specifies the identifier of the presentation slide. This element MUST be present.

Theme: A **Theme** complex element (section [2.2.4.49](#)) that specifies the theme for the presentation slide. This element MUST be present.

ThumbnailViewElement: A **ViewElement** complex element (section [2.2.4.51](#)) that specifies the thumbnail of the presentation slide. This element MUST be present.

ViewElems: An **ArrayOfViewElement** complex element (section [2.2.4.14](#)) that specifies the **ViewElement** elements for slide content of the presentation slide. This element MUST be present.

2.2.4.30 EditSlideChanges

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

The **EditSlideChanges** complex type specifies the content of the presentation slide.

The content of this element is modified as a result of a request to the protocol server.

```
<xs:complexType name="EditSlideChanges" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Comments" nillable="true" type="tns:ArrayOfComment"/>
  </xs:sequence>
</xs:complexType>
```

```

    <xs:element xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
minOccurs="0" name="DelElems" nillable="true" type="q3:ArrayOfunsignedInt"/>
    <xs:element minOccurs="0" name="EditAnims" nillable="true"
type="tns2:ArrayOfEditAnimInfo"/>
    <xs:element minOccurs="0" name="EditTransition" nillable="true"
type="tns2:EditTransitionInfo"/>
    <xs:element minOccurs="0" name="Elems" nillable="true" type="tns2:ArrayOfViewElement"/>
    <xs:element minOccurs="0" name="HasAnimationOrTransition" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Hidden" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Id" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="ModifiedShapes" nillable="true"
type="tns2:ArrayOfEditShape"/>
    <xs:element minOccurs="0" name="Notes" nillable="true" type="tns2:Notes"/>
    <xs:element minOccurs="0" name="Refresh" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Thumb" nillable="true" type="tns2:ViewElement"/>
    <xs:element minOccurs="0" name="Title" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

```

Comments: An **ArrayOfComment** complex element (section [2.2.4.1](#)) that specifies comments for the presentation slide.

DelElems: Reserved. MUST be ignored.

EditAnims: An **ArrayOfEditAnimInfo** complex element (section [2.2.4.3](#)) that specifies the list of animations on the slide.

EditTransition: An **EditTransitionInfo** complex element (section [2.2.4.32](#)) that specifies the slide transition.

Elems: An **ArrayOfViewElement** complex element (section [2.2.4.14](#)) that specifies the **ViewElement** complex elements (section [2.2.4.51](#)) for slide content on the presentation slide.

HasAnimationOrTransition:

Hidden: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that specifies whether the presentation slide is hidden.

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

ModifiedShapes: An **ArrayOfEditShape** complex element (section [2.2.4.5](#)) that specifies the shapes modified.

Notes: A **Notes** complex element (section [2.2.4.37](#)) that specifies the presentation notes for the presentation slide.

Refresh: A **boolean** that specifies whether the presentation slide has outdated **ViewElement** elements.

Thumb: A **ViewElement** element that specifies the thumbnail of the presentation slide.

Title: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the title of the presentation slide.

2.2.4.31 EditSlideInfoFe

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

The **EditSlideInfoFe** complex type specifies information about a presentation slide.

```

<xs:complexType name="EditSlideInfoFe" xmlns:xs="http://www.w3.org/2001/XMLSchema">

```

```

<xs:sequence>
  <xs:element minOccurs="0" name="HasAnimationOrTransition" type="xs:boolean"/>
  <xs:element minOccurs="0" name="Hidden" type="xs:boolean"/>
  <xs:element minOccurs="0" name="Id" type="xs:unsignedInt"/>
  <xs:element minOccurs="0" name="Thumb" nillable="true" type="tns2:ViewElement"/>
  <xs:element minOccurs="0" name="Title" nillable="true" type="xs:string"/>
</xs:sequence>
</xs:complexType>

```

HasAnimationOrTransition:

Hidden: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that specifies whether the presentation slide is hidden.

Id: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies an identifier for the presentation slide.

Thumb: A **ViewElement** complex element (section [2.2.4.51](#)) that specifies the thumbnail of the presentation slide.

Title: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the title of the presentation slide.

2.2.4.32 EditTransitionInfo

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

The **EditTransitionInfo** complex type [<31>](#) specifies a slide transition.

```

<xs:complexType name="EditTransitionInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Direction" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Duration" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="IsManualAdvance" type="xs:boolean"/>
    <xs:element minOccurs="0" name="TransitionType" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>

```

Direction: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the direction of the transition.

Duration: An **unsignedInt** that specifies the duration of the transition in milliseconds.

IsManualAdvance: Reserved. MUST be ignored.

TransitionType: An **unsignedInt** that specifies the type of the transition.

2.2.4.33 FontSizeMapping

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

Reserved; this element MUST be ignored.

```

<xs:complexType name="FontSizeMapping" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Size1" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Size2" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Size3" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Size4" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Size5" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>

```

```

    <xs:element minOccurs="0" name="Size6" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Size7" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>

```

Size1: Reserved. MUST be ignored.

Size2: Reserved. MUST be ignored.

Size3: Reserved. MUST be ignored.

Size4: Reserved. MUST be ignored.

Size5: Reserved. MUST be ignored.

Size6: Reserved. MUST be ignored.

Size7: Reserved. MUST be ignored.

2.2.4.34 **Html**

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

The **Html** complex type specifies HTML content.

```

<xs:complexType name="Html" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Text" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

```

Text: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the HTML content.

2.2.4.35 **MainMasterFe**

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

The **MainMasterFe** complex type specifies a main master slide.

```

<xs:complexType name="MainMasterFe" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="ContentMasters" nillable="true"
type="tns:ArrayOfContentMasterFe"/>
    <xs:element minOccurs="0" name="Id" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="InContent" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="ThemeFamilyId" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="VariantIndex" type="xs:int"/>
  </xs:sequence>
</xs:complexType>

```

ContentMasters: An **ArrayOfContentMasterFe** complex element (section [2.2.4.2](#)) that specifies slide layouts.

Id: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the identifier of the main master slide.

InContent: Reserved. MUST be ignored.

Name: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the name of the main master slide.

ThemeFamilyId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the theme of the main master slide.

VariantIndex: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the theme variant of the main master slide.

2.2.4.36 MainMasterResourcesFe

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

Reserved. MUST be ignored.

```
<xs:complexType name="MainMasterResourcesFe" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="LineStyle" nillable="true"
type="tns2:ArrayOfStyleGalleryItem"/>
    <xs:element minOccurs="0" name="MasterId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="ShapeStyles" nillable="true"
type="tns2:ArrayOfStyleGalleryItem"/>
  </xs:sequence>
</xs:complexType>
```

LineStyle: Reserved. MUST be ignored.

MasterId: Reserved. MUST be ignored.

ShapeStyles: Reserved. MUST be ignored.

2.2.4.37 Notes

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

The **Notes** complex type specifies the notes of a presentation slide.

```
<xs:complexType name="Notes" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Html" nillable="true" type="tns2:Html"/>
    <xs:element minOccurs="0" name="IsPrompt" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
```

Html: An **Html** complex element (section [2.2.4.34](#)) that specifies the content of the notes.

IsPrompt: A **boolean** ([XMLSCHEMA2] section 3.2.2) that specifies whether the notes is empty.

2.2.4.38 Point

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

The **Point** complex type specifies a two dimensional point.

```
<xs:complexType name="Point" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="X" type="xs:int"/>
    <xs:element minOccurs="0" name="Y" type="xs:int"/>
  </xs:sequence>
```

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

X: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the x coordinate. This element **MUST** be present.

Y: An **int** that specifies the y coordinate. This element **MUST** be present.

2.2.4.39 PptViewingService.PrintResult

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

The **PptViewingService.PrintResult** complex type [<32>](#) specifies the result of a print operation.

```
<xs:complexType name="PptViewingService.PrintResult"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="PrintUrl" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

PrintUrl: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) element that specifies the URL for getting a printable version of the presentation.

2.2.4.40 PresetSchemeColorMapping

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

The **PresetSchemeColorMapping** complex type [<33>](#) specifies a set of colors in the theme.

```
<xs:complexType name="PresetSchemeColorMapping" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Accent1" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent2" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent3" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent4" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent5" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent6" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Background1" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Background2" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Text1" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Text2" nillable="true" type="tns2:ThemeColor"/>
  </xs:sequence>
</xs:complexType>
```

Accent1: A **ThemeColor** complex element (section [2.2.4.50](#)) that defines the first accent color in the theme.

Accent2: A **ThemeColor** element that defines the second accent color in the theme.

Accent3: A **ThemeColor** element that defines the third accent color in the theme.

Accent4: A **ThemeColor** element that defines the fourth accent color in the theme.

Accent5: A **ThemeColor** element that defines the fifth accent color in the theme.

Accent6: A **ThemeColor** element that defines the sixth accent color in the theme.

Background1: A **ThemeColor** element that defines the first background color in the theme.

Background2: A **ThemeColor** element that defines the second background color in the theme.

Text1: A **ThemeColor** element that defines the first text color in the theme.

Text2: A **ThemeColor** element that defines the second text color in the theme.

2.2.4.41 Rectangle

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

The **Rectangle** complex type specifies a rectangle.

```
<xs:complexType name="Rectangle" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Bottom" type="xs:int"/>
    <xs:element minOccurs="0" name="Left" type="xs:int"/>
    <xs:element minOccurs="0" name="Right" type="xs:int"/>
    <xs:element minOccurs="0" name="Top" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

Bottom: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the location of the bottom of the rectangle. This element **MUST** be present.

Left: An **int** that specifies the location of the left side of the rectangle. This element **MUST** be present.

Right: An **int** that specifies the location of the right side of the rectangle. This element **MUST** be present.

Top: An **int** that specifies the location of the top of the rectangle. This element **MUST** be present.

2.2.4.42 ServiceError

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

The **ServiceError** complex type specifies error information.

This information is returned by the protocol server to a protocol client.

```
<xs:complexType name="ServiceError" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element
      xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared"
      minOccurs="0" name="Code" type="tns:ErrorCode"/>
    <xs:element minOccurs="0" name="Message" nillable="true" type="xs:string"/>
    <xs:element
      xmlns:tns1="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.Interface"
      minOccurs="0" name="RecommendedActions" type="tns1:ClientActions"/>
    <xs:element minOccurs="0" name="Title" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Type" type="tns2:ServiceErrorType"/>
  </xs:sequence>
</xs:complexType>
```

Code: Reserved. **MUST** be ignored.

Message: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the error message description. This element **MUST** be present. The string length **MUST** be greater than zero, if the **Type** element has a value of **ApplicationError** (see section [2.2.5.6](#)).

RecommendedActions: Reserved. **MUST** be ignored.

Title: A **string** that specifies the error title. This element **MUST** be present. The string length **MUST** be greater than zero, if the **Type** element has a value of **ApplicationError**.

Type: A **ServiceErrorType** simple type (section 2.2.5.6) that specifies the error type. This element **MUST** be present.

2.2.4.43 ServiceResult

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

The **ServiceResult** complex type 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" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Error" nillable="true" type="tns2:ServiceError"/>
    <xs:element minOccurs="0" name="Result" nillable="true" type="xs:anyType"/>
  </xs:sequence>
</xs:complexType>
```

Error: An optional **ServiceError** complex element (section [2.2.4.42](#)) that specifies an error result of a protocol message response. This element **MUST NOT** be empty if the **Result** element is empty. Otherwise, this element **MUST** be empty.

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

2.2.4.44 SlideID

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

A complex type that **SHOULD** [<34>](#) identify a presentation slide.

```
<xs:complexType name="SlideID" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Cid" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Sid" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
```

Cid: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the creation identifier of a presentation slide.

Sid: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the identifier of a presentation slide.

2.2.4.45 SlideIdList

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

The **SlideIdList** complex type specifies a list of identifiers for presentation slides.

```
<xs:complexType name="SlideIdList" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
```

```

    <xs:element xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
      minOccurs="0" name="SlideIds" nillable="true" type="q3:ArrayOfunsignedInt"/>
  </xs:sequence>
</xs:complexType>

```

SlideIds: An **ArrayOfunsignedInt** complex element (section [2.2.4.13](#)) that specifies the list of identifiers.

2.2.4.46 SlideUpdateInfo

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

Reserved. MUST be ignored.

```

<xs:complexType name="SlideUpdateInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="leid" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="lsu" type="xs:dateTime"/>
    <xs:element minOccurs="0" name="sid" type="tns2:SlideID"/>
  </xs:sequence>
</xs:complexType>

```

leid: Reserved. MUST be ignored.

lsu: Reserved. MUST be ignored.

sid: Reserved. MUST be ignored.

2.2.4.47 SmartArtInfo

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

The **SmartArtInfo** complex type specifies information about a SmartArt diagram.

```

<xs:complexType name="SmartArtInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="fReverseChecked" type="xs:boolean"/>
    <xs:element minOccurs="0" name="fReverseEnabled" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>

```

fReverseChecked: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that specifies whether the SmartArt diagram is reversed.

fReverseEnabled: A **boolean** that specifies whether the SmartArt diagram can be reversed.

2.2.4.48 StyleGalleryItem

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

The **StyleGalleryItem** complex type [<35>](#) specifies a shape style.

```

<xs:complexType name="StyleGalleryItem" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Icon" nillable="true" type="tns2:ViewElement"/>
    <xs:element minOccurs="0" name="StyleId" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>

```

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

Icon: A **ViewElement** complex element (section [2.2.4.51](#)) that specifies the style of the shape. This element **MUST** be present.

StyleId: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the type of the shape style. This element **MUST** be present.

2.2.4.49 Theme

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

The **Theme** complex type specifies information about a theme.

```
<xs:complexType name="Theme" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Accent1" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent2" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent3" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent4" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent5" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent6" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Dark1" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Dark2" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Light1" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Light2" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="PresetSchemeColorMapping" nillable="true"
type="tns2:PresetSchemeColorMapping"/>
  </xs:sequence>
</xs:complexType>
```

Accent1: A **ThemeColor** complex element (section [2.2.4.50](#)) that defines the first accent color in the theme.

Accent2: A **ThemeColor** element that defines the second accent color in the theme.

Accent3: A **ThemeColor** element that defines the third accent color in the theme.

Accent4: A **ThemeColor** element that defines the fourth accent color in the theme.

Accent5: A **ThemeColor** element that defines the fifth accent color in the theme

Accent6: A **ThemeColor** element that defines the sixth accent color in the theme.

Dark1: A **ThemeColor** element that defines a dark color in the theme.

Dark2: A **ThemeColor** element that defines a second dark color in the theme.

Light1: A **ThemeColor** element that defines a light color in the theme.

Light2: A **ThemeColor** element that defines a second light color in the theme.

PresetSchemeColorMapping: A **PresetSchemeColorMapping** complex element (section [2.2.4.40](#)) that defines a set of colors in the theme.

2.2.4.50 ThemeColor

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

The **ThemeColor** complex type specifies a color using the red, green, blue RGB color model.

Red, green, and blue are each expressed as an unsigned byte from 0 for no intensity to 255 for full intensity.

```
<xs:complexType name="ThemeColor" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="b" type="xs:unsignedByte"/>
    <xs:element minOccurs="0" name="g" type="xs:unsignedByte"/>
    <xs:element minOccurs="0" name="r" type="xs:unsignedByte"/>
  </xs:sequence>
</xs:complexType>
```

b: An **unsignedByte** ([\[XMLSCHEMA2\]](#) section 3.3.24) that specifies the blue component of the color. This element MUST be present.

g: An **unsignedByte** that specifies the green component of the color. This element MUST be present.

r: An **unsignedByte** that specifies the red component of the color. This element MUST be present.

2.2.4.51 ViewElement

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

The **ViewElement** complex type specifies a visual representation of presentation slide content.

```
<xs:complexType name="ViewElement" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Alt" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Clip" nillable="true" type="tns2:ClippingInfo"/>
    <xs:element minOccurs="0" name="Html" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Id" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="ImgId" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Position" nillable="true" type="tns2:Point"/>
    <xs:element minOccurs="0" name="Z" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
```

Alt: Reserved. MUST be ignored.

Clip: A **ClippingInfo** complex element (section [2.2.4.16](#)) that specifies the region within the image. This element MUST be present.

Html: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies text content. This element MUST be present, if and only if **ImgId** is not present.

Id: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the identifier. This element MUST be present.

ImgId: A **string** that specifies the URL of the image. [<36>](#)

Position: A **Point** complex element (section [2.2.4.38](#)) that specifies the position.

Z: An **unsignedInt** that specifies the z-order.

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. MUST be ignored. The client and the server MUST NOT use this as the type of an element, and the client and the server MUST ignore it, if receiving an element of this type.
ClientActions	Reserved. MUST be ignored.
duration	Reserved. MUST be ignored. The client and the server MUST NOT use this as the type of an element, and the client and the server MUST ignore it, if receiving an element of this type.
ErrorCode	Reserved. MUST be ignored.
guid	Reserved. MUST be ignored. The client and the server MUST NOT use this as the type of an element, and the client and the server MUST ignore it, if receiving an element of this type.
ServiceErrorType	A simple type that specifies an enumeration of a set of protocol errors returned by the protocol server to the protocol client.

2.2.5.1 char

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

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

```
<xs:simpleType name="char" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:restriction base="xs:int"/>
</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" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <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 the **ClientActions** simple type.

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. MUST be ignored. The client and the server MUST NOT use this as the type of an element, and the client and the server MUST ignore it, if receiving an element of this type.

```
<xs:simpleType name="duration" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <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 ErrorCode

Namespace:

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

Reserved. MUST be ignored.

```
<xs:simpleType name="ErrorCode" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:restriction base="xs:string">
    <xs:enumeration value="ErrorRESERVEDSuccess"/>
    <xs:enumeration value="ErrorRESERVEDUnknown"/>
    <xs:enumeration value="ErrorCannotOpenDocumentFileType"/>
    <xs:enumeration value="ErrorCannotOpenDocumentIRM"/>
    <xs:enumeration value="ErrorCannotOpenDocumentPassword"/>
    <xs:enumeration value="ErrorCannotOpenDocument"/>
    <xs:enumeration value="ErrorCannotInstallFonts"/>
    <xs:enumeration value="ErrorCannotConvertModifyPassword"/>
    <xs:enumeration value="ErrorCannotComposeSlideParse"/>
    <xs:enumeration value="ErrorCannotComposeSlideUnexpected"/>
    <xs:enumeration value="ErrorWacConversionFailure"/>
    <xs:enumeration value="ErrorResourceNotGenerated"/>
    <xs:enumeration value="ErrorGenericSlideCorruption"/>
    <xs:enumeration value="ErrorFileNotFound"/>
    <xs:enumeration value="ErrorGeneralUnknown"/>
    <xs:enumeration value="ErrorFailedToOpenDocument"/>
    <xs:enumeration value="ErrorFailedToRetrieveResource"/>
    <xs:enumeration value="ErrorFailedToRenderDocument"/>
    <xs:enumeration value="ErrorFailedToRenderSlide"/>
    <xs:enumeration value="ErrorFailedToSaveDocument"/>
    <xs:enumeration value="ErrorCannotSaveDocument"/>
    <xs:enumeration value="ErrorInvalidCommand"/>
    <xs:enumeration value="ErrorDocVersionMismatch"/>
    <xs:enumeration value="ErrorCannotAcquirePptEditLock"/>
  </xs:restriction>
</xs:simpleType>
```

```

<xs:enumeration value="ErrorRequiresCheckout"/>
<xs:enumeration value="ErrorLockedForOfflineCheckout"/>
<xs:enumeration value="ErrorLockedByAnotherUser"/>
<xs:enumeration value="ErrorAccessDenied"/>
<xs:enumeration value="ErrorNoLicense"/>
<xs:enumeration value="ErrorRequestTooBig"/>
<xs:enumeration value="ErrorTextTooBig"/>
<xs:enumeration value="ErrorTryAgain"/>
<xs:enumeration value="ErrorUnexpected"/>
<xs:enumeration value="ErrorCannotOpenDocumentMacro"/>
<xs:enumeration value="ErrorCannotOpenDocumentFinal"/>
<xs:enumeration value="ErrorCannotOpenDocumentSigned"/>
<xs:enumeration value="ErrorCannotOpenEmbeddedFonts"/>
<xs:enumeration value="ErrorNoAvailableEditAppServer"/>
<xs:enumeration value="ErrorServiceUnavailable"/>
<xs:enumeration value="ErrorPptEditExistingExclusiveLock"/>
<xs:enumeration value="ErrorPptEditCheckedOutByAnother"/>
<xs:enumeration value="ErrorCannotOpenDocumentGatekeeperFailure"/>
<xs:enumeration value="ErrorEditWorkerTimeout"/>
<xs:enumeration value="ErrorEditSlideCorruption"/>
<xs:enumeration value="ErrorServiceBusy"/>
<xs:enumeration value="ErrorClipartCannotConnect"/>
<xs:enumeration value="ErrorClipartCannotDownload"/>
<xs:enumeration value="ErrorMediaNotEnabled"/>
<xs:enumeration value="ErrorUnsupportedMediaFormat"/>
<xs:enumeration value="ErrorLinkedMediaNotEnabled"/>
<xs:enumeration value="ErrorUnsupportedLinkTarget"/>
<xs:enumeration value="ErrorEmbeddedMediaNotEnabled"/>
<xs:enumeration value="ErrorEmbeddedMediaExceedsMaxSize"/>
<xs:enumeration value="ErrorTranscodingNotEnabled"/>
<xs:enumeration value="ErrorMediaUnavailable"/>
<xs:enumeration value="ErrorCannotOpenODPDocumentPassword"/>
<xs:enumeration value="ErrorLocalChangeLostDueToCoauthorEdit"/>
<xs:enumeration value="ErrorNumberOfCoauthorsReachedMax"/>
<xs:enumeration value="ErrorCannotAcquirePptEditSharedLock"/>
<xs:enumeration value="ErrorInProgress"/>
<xs:enumeration value="ErrorImageTooLarge"/>
<xs:enumeration value="ErrorImageWrongFormat"/>
<xs:enumeration value="ErrorImageNoAccess"/>
<xs:enumeration value="ErrorWacItemRetrievalDocumentOutputNotFound"/>
<xs:enumeration value="ErrorWacItemRetrievalAccessDenied"/>
<xs:enumeration value="ErrorWacItemRetrievalCommunicationErrorStreamDirty"/>
<xs:enumeration value="ErrorWacItemRetrievalCommunicationErrorStreamClean"/>
<xs:enumeration value="ErrorWacItemRetrievalUnknownError"/>
<xs:enumeration value="ErrorWacConversionManagerDoesNotSupportThisRequest"/>
<xs:enumeration value="ErrorWacConversionCannotFindSourceDocument"/>
<xs:enumeration value="ErrorWacConversionWorkerException"/>
<xs:enumeration value="ErrorWacConversionWorkerCrashed"/>
<xs:enumeration value="ErrorWacConversionWorkerHung"/>
<xs:enumeration value="ErrorWacConversionUnknownResponse"/>
<xs:enumeration value="ErrorWacConversionUnknownError"/>
<xs:enumeration value="ErrorWacConversionFirstServerResult"/>
<xs:enumeration value="ErrorCannotOpenDocumentWrongExt"/>
<xs:enumeration value="ErrorCannotOpenDocumentStgDocFileCorrupt"/>
<xs:enumeration value="ErrorCannotOpenDocumentStgFileAlreadyExists"/>
<xs:enumeration value="ErrorCannotOpenDocumentMmcfPackagePartNotFound"/>
<xs:enumeration value="ErrorCannotOpenDocumentArchiveError"/>
<xs:enumeration value="ErrorCannotPastePictureCannotCreateUrl"/>
<xs:enumeration value="ErrorCannotPastePictureInvalidResponse"/>
<xs:enumeration value="ErrorCannotPastePictureNotFound"/>
<xs:enumeration value="ErrorCannotPastePictureResponseEmpty"/>
<xs:enumeration value="ErrorCannotPastePictureUnexpectedStatusCode"/>
<xs:enumeration value="ErrorCannotPastePictureUnexpectedWebException"/>
<xs:enumeration value="ErrorRESERVEDLastError"/>
</xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for the **ErrorCode** simple type.

Value	Meaning
ErrorRESERVEDSuccess	Reserved. MUST be ignored.
ErrorRESERVEDUnknown	Reserved. MUST be ignored.
ErrorCannotOpenDocumentFileType	Reserved. MUST be ignored.
ErrorCannotOpenDocumentIRM	Reserved. MUST be ignored.
ErrorCannotOpenDocumentPassword	Reserved. MUST be ignored.
ErrorCannotOpenDocument	Reserved. MUST be ignored.
ErrorCannotInstallFonts	Reserved. MUST be ignored.
ErrorCannotConvertModifyPassword	Reserved. MUST be ignored.
ErrorCannotComposeSlideParse	Reserved. MUST be ignored.
ErrorCannotComposeSlideUnexpected	Reserved. MUST be ignored.
ErrorWacConversionFailure	Reserved. MUST be ignored.
ErrorResourceNotGenerated	Reserved. MUST be ignored.
ErrorGenericSlideCorruption	Reserved. MUST be ignored.
ErrorFileNotFound	Reserved. MUST be ignored.
ErrorGeneralUnknown	Reserved. MUST be ignored.
ErrorFailedToOpenDocument	Reserved. MUST be ignored.
ErrorFailedToRetrieveResource	Reserved. MUST be ignored.
ErrorFailedToRenderDocument	Reserved. MUST be ignored.
ErrorFailedToRenderSlide	Reserved. MUST be ignored.
ErrorFailedToSaveDocument	Reserved. MUST be ignored.
ErrorCannotSaveDocument	Reserved. MUST be ignored.
ErrorInvalidCommand	Reserved. MUST be ignored.
ErrorDocVersionMismatch	Reserved. MUST be ignored.
ErrorCannotAcquirePptEditLock	Reserved. MUST be ignored.
ErrorRequiresCheckout	Reserved. MUST be ignored.
ErrorLockedForOfflineCheckout	Reserved. MUST be ignored.
ErrorLockedByAnotherUser	Reserved. MUST be ignored.
ErrorAccessDenied	Reserved. MUST be ignored.
ErrorNoLicense	Reserved. MUST be ignored.
ErrorRequestTooBig	Reserved. MUST be ignored.
ErrorTextTooBig	Reserved. MUST be ignored.
ErrorTryAgain	Reserved. MUST be ignored.

Value	Meaning
ErrorUnexpected	Reserved. MUST be ignored.
ErrorCannotOpenDocumentMacro	Reserved. MUST be ignored.
ErrorCannotOpenDocumentFinal	Reserved. MUST be ignored.
ErrorCannotOpenDocumentSigned	Reserved. MUST be ignored.
ErrorCannotOpenEmbeddedFonts	Reserved. MUST be ignored.
ErrorNoAvailableEditAppServer	Reserved. MUST be ignored.
ErrorServiceUnavailable	Reserved. MUST be ignored.
ErrorPptEditExistingExclusiveLock	Reserved. MUST be ignored.
ErrorPptEditCheckedOutByAnother	Reserved. MUST be ignored.
ErrorCannotOpenDocumentGatekeeperFailure	Reserved. MUST be ignored.
ErrorEditWorkerTimeout	Reserved. MUST be ignored.
ErrorEditSlideCorruption	Reserved. MUST be ignored.
ErrorServiceBusy	Reserved. MUST be ignored.
ErrorClipartCannotConnect	Reserved. MUST be ignored.
ErrorClipartCannotDownload	Reserved. MUST be ignored.
ErrorMediaNotEnabled	Reserved. MUST be ignored.
ErrorUnsupportedMediaFormat	Reserved. MUST be ignored.
ErrorLinkedMediaNotEnabled	Reserved. MUST be ignored.
ErrorUnsupportedLinkTarget	Reserved. MUST be ignored.
ErrorEmbeddedMediaNotEnabled	Reserved. MUST be ignored.
ErrorEmbeddedMediaExceedsMaxSize	Reserved. MUST be ignored.
ErrorTranscodingNotEnabled	Reserved. MUST be ignored.
ErrorMediaUnavailable	Reserved. MUST be ignored.
ErrorCannotOpenODPDDocumentPassword	Reserved. MUST be ignored.
ErrorLocalChangeLostDueToCoauthorEdit	Reserved. MUST be ignored.
ErrorNumberOfCoauthorsReachedMax	Reserved. MUST be ignored.
ErrorCannotAcquirePptEditSharedLock	Reserved. MUST be ignored.
ErrorInProgress	Reserved. MUST be ignored.
ErrorImageTooLarge	Reserved. MUST be ignored.
ErrorImageWrongFormat	Reserved. MUST be ignored.
ErrorImageNoAccess	Reserved. MUST be ignored.
ErrorWacItemRetrievalDocumentOutputNotFound	Reserved. MUST be ignored.

Value	Meaning
ErrorWacItemRetrievalAccessDenied	Reserved. MUST be ignored.
ErrorWacItemRetrievalCommunicationErrorStreamDirty	Reserved. MUST be ignored.
ErrorWacItemRetrievalCommunicationErrorStreamClean	Reserved. MUST be ignored.
ErrorWacItemRetrievalUnknownError	Reserved. MUST be ignored.
ErrorWacConversionManagerDoesNotSupportThisRequest	Reserved. MUST be ignored.
ErrorWacConversionCannotFindSourceDocument	Reserved. MUST be ignored.
ErrorWacConversionWorkerException	Reserved. MUST be ignored.
ErrorWacConversionWorkerCrashed	Reserved. MUST be ignored.
ErrorWacConversionWorkerHung	Reserved. MUST be ignored.
ErrorWacConversionUnknownResponse	Reserved. MUST be ignored.
ErrorWacConversionUnknownError	Reserved. MUST be ignored.
ErrorWacConversionFirstServerResult	Reserved. MUST be ignored.
ErrorCannotOpenDocumentWrongExt	Reserved. MUST be ignored.
ErrorCannotOpenDocumentStgDocFileCorrupt	Reserved. MUST be ignored.
ErrorCannotOpenDocumentStgFileAlreadyExists	Reserved. MUST be ignored.
ErrorCannotOpenDocumentMmcfPackagePartNotFound	Reserved. MUST be ignored.
ErrorCannotOpenDocumentArchiveError	Reserved. MUST be ignored.
ErrorCannotPastePictureCannotcreateUrl	Reserved. MUST be ignored.
ErrorCannotPastePictureInvalidResponse	Reserved. MUST be ignored.
ErrorCannotPastePictureNotFound	Reserved. MUST be ignored.
ErrorCannotPastePictureResponseEmpty	Reserved. MUST be ignored.
ErrorCannotPastePictureUnexpectedStatusCode	Reserved. MUST be ignored.
ErrorCannotPastePictureUnexpectedWebException	Reserved. MUST be ignored.
ErrorRESERVEDLastError	Reserved. MUST be ignored.

2.2.5.5 guid

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

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

```
<xs:simpleType name="guid" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <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:restriction>
</xs:simpleType>
```

2.2.5.6 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" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <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 the **ServiceErrorType** simple type.

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	Reserved. MUST be ignored.
SessionFull	Reserved. MUST be ignored.

2.2.6 Attributes

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

2.2.7 Groups

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

2.2.8 Attribute Groups

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

3 Protocol Details

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

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

3.1 Server Details

The following diagram describes the communication between the protocol client and the protocol server.

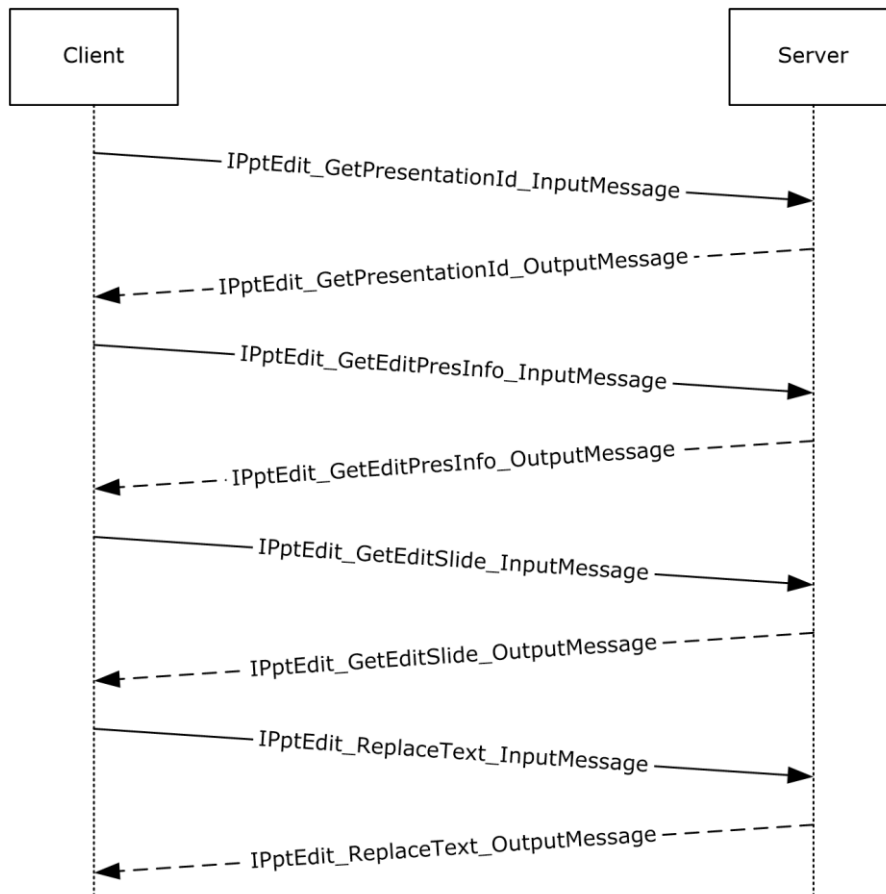


Figure 2: Sample communication between protocol client and protocol server

The protocol client sends the **IPptEdit_GetPresentationId_InputMessage SOAP message** as specified in section [3.1.4.27](#) to obtain a valid identifier for a presentation within an **IPptEdit_GetPresentationId_OutputMessage**.

The protocol client sends an **IPptEdit_GetEditPresInfo_InputMessage** message and the protocol server responds with an **IPptEdit_GetEditPresInfo_OutputMessage** containing information about the requested presentation.

The protocol client sends one or more **IPptEdit_GetEditSlide_InputMessage** messages and the protocol server responds with an **IPptEdit_GetSlide_OutputMessage** containing information about the requested presentation slide.

The protocol client sends one or more **IPptEdit_ReplaceText_InputMessage** messages and the protocol server responds with an **IPptEdit_ReplaceText_OutputMessage** containing information about the result of the operation.

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.

3.1.2 Timers

None.

3.1.3 Initialization

The protocol server MUST expose its Web methods at the following URLs.

The URL which the client sends requests to modify presentation content SHOULD <37> conform to the following structure: *base URL/p/ppt/edit.svc*. This is the minimal required structure. Case-sensitivity is specific to the protocol server implementation.

The URL which the client sends requests to modify image content SHOULD <38> conform to the following structure: *base URL/p/pptInsertPicture.ashx*. This is the minimal required structure. Case-sensitivity is specific to the protocol server implementation.

3.1.4 Message Processing Events and Sequencing Rules

The message processing events and sequencing rules are listed in this section.

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

Operation	Description
AddComment	The AddComment operation SHOULD <39> be used to insert a comment into a presentation.
ApplyShapeFill	The ApplyShapeFill operation SHOULD <40> be used to modify the color of a shape in a presentation.
ApplyShapeOutlineColor	The ApplyShapeOutlineColor operation SHOULD <41> be used to modify the color of the outline of a shape in a presentation.
ApplyShapeOutlineDashStyle	The ApplyShapeOutlineDashStyle operation SHOULD <42> be used to modify the dash type of the outline of a shape.
ApplyShapeOutlineEndStyle	The ApplyShapeOutlineEndStyle operation SHOULD <43> be used to modify the end style of a line in a presentation.

Operation	Description
ApplyShapeOutlineWidth	The ApplyShapeOutlineWidth operation SHOULD <44> be used to modify the width of the outline of a shape.
ApplyShapeStyle	The ApplyShapeStyle operation SHOULD <45> be used to apply a style to a shape in a presentation.
ApplyTheme	The ApplyTheme operation SHOULD <46> be used to change the theme applied to a presentation.
ArrangeShape	The ArrangeShape operation SHOULD <47> be used to rearrange a shape in a presentation.
ChangeLayout	The ChangeLayout operation is used to change the layout of a slide.
ChangePictureStyle	The ChangePictureStyle operation is used to change the style applied to a picture.
ChangeSmartArtColor	The ChangeSmartArtColor operation is used to change the color transform applied to a SmartArt diagram.
ChangeSmartArtLayout	The ChangeSmartArtLayout operation is used to change the layout applied to a SmartArt diagram.
ChangeSmartArtStyle	The ChangeSmartArtStyle operation is used to change the style applied to a SmartArt diagram.
ClearPlaceholder	The ClearPlaceholder is used to delete a shape in a presentation.
DeleteComment	The DeleteComment operation SHOULD <48> be used to delete a comment from a presentation.
DeleteSlide	The DeleteSlide operation is used to delete a presentation slide from a presentation.
DuplicateShape	The DuplicateShape operation SHOULD <49> be used to duplicate a shape in a presentation.
DuplicateSlide	The DuplicateSlide operation is used to duplicate a presentation slide.
EditComment	The EditComment operation SHOULD <50> be used to change the contents of a comment in a presentation
FlipShape	The FlipShape operation SHOULD <51> be used to flip a shape vertically or horizontally in a presentation.
GetCoauthUpdates	The GetCoauthUpdates operation SHOULD <52> be used to retrieve co-authoring state for a presentation.
GetEditPresInfo	The GetEditPresInfo operation is used to retrieve information about a presentation.
GetEditPresInfoGetEditSlideById	The GetEditPresInfoGetEditSlideById operation SHOULD <53> be used to retrieve information about a presentation.
GetEditPresResources	Reserved. MUST not be called.
GetEditSlide	The GetEditSlide operation is used to retrieve information about a presentation slide.
GetPresentationId	The GetPresentationId operation is used to retrieve an identifier for a presentation.
InsertClipart	The InsertClipart operation SHOULD <54> be used to insert a picture

Operation	Description
	into a presentation.
InsertShape	The InsertShape operation SHOULD <55> be used to insert a shape in a presentation.
InsertSlide	The InsertSlide operation is used to insert a presentation slide into a presentation.
InsertSmartArt	The InsertSmartArt operation is used to insert a SmartArt diagram into a presentation
LogULS	Reserved. MUST not be called.
MoveComment	The MoveComment operation SHOULD <56> be used to change the position of a comment in a presentation.
MoveShape	The MoveShape operation SHOULD <57> be used to move a shape in a presentation.
MoveSlide	The MoveSlide operation is used to move a presentation slide in a presentation.
PastePicture	The PastePicture operation is used to paste a picture into a presentation.
Print	A Print operation SHOULD <58> be used to print a presentation.
Redo	The Redo operation is used to redo the last operation in a presentation that is undone.
RemoveShapeFill	The RemoveShapeFill operation SHOULD <59> be used to remove the color of a shape in a presentation.
RemoveShapeOutline	The RemoveShapeOutline operation SHOULD <60> be used to remove the outline of a shape in a presentation.
ReorderAnimation	The ReorderAnimation operation SHOULD <61> be used to reorder the animation of a shape in a presentation.
ReplaceNotes	The ReplaceNotes operation is used to replace presentation notes in a presentation.
ReplaceText	The ReplaceText operation is used to replace text in a shape in a presentation.
ResetPicture	The ResetPicture operation SHOULD <62> be used to remove all customizations on a picture in a presentation.
ResetSmartArt	The ResetSmartArt operation is used to remove all customizations on a SmartArt diagram in a presentation.
ResizeShape	The ResizeShape operation SHOULD <63> be used to resize a shape in a presentation.
ReverseSmartArt	The ReverseSmartArt operation is used to reverse the direction of a SmartArt diagram in a presentation.
RotateShape	The RotateShape operation SHOULD <64> be used to rotate a shape in a presentation.
SaveAndClose	The SaveAndClose operation is used to save the contents of a presentation and unlock it.
SetAnimation	The SetAnimation operation SHOULD <65> be used to apply an

Operation	Description
	animation to a shape in a presentation.
SetShapeAlignment	The SetShapeAlignment operation is used to set the alignment of a shape.
SetShapeBold	The SetShapeBold operation is used to make the text in a shape bold.
SetShapeBullet	The SetShapeBullet operation is used to make the text in a shape into a bulleted list.
SetShapeFontColor	The SetShapeFontColor operation is used to set the font color of a shape.
SetShapeFontName	The SetShapeFontName operation is used to set the font name for text for a shape.
SetShapeFontSize	The SetShapeFontSize operation is used to set the font size for text for a shape.
SetShapeItalic	The SetShapeItalic operation is used to italicize the text for a shape.
SetShapeTextDirection	The SetShapeTextDirection operation is used to set the text direction for a shape.
SetShapeUnderline	The SetShapeUnderline operation is used to underline text in a shape.
SetTransition	The SetTransition operation SHOULD 66 be used to apply a transition to a slide in a presentation.
SetTransitionApplyAll	The SetTransitionApplyAll operation SHOULD 67 be used to copy a transition from one slide and apply it to all other slides in a presentation.
SetWordArt	Reserved. MUST not be called.
ShapeFormatPainting	The ShapeFormatPainting operation SHOULD 68 be used to copy formatting from one shape and apply it to another shape in the presentation.
ShowHideSlide	The ShowHideSlide operation is used to show or hide a presentation slide.
Undo	The Undo operation is used to undo the last operation in a presentation that is not undone.
UngroupShape	The UngroupShape operation SHOULD 69 be used to ungroup a shape in a presentation.
UpdateCanary	Reserved. MUST be ignored.

3.1.4.1 AddComment

The **AddComment** operation SHOULD [70](#) be used to insert a comment into a presentation.

The following is the WSDL port type specification of the **AddComment WSDL operation**.

```
<wsdl:operation name="AddComment" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/AddComment" message="tns4:IPptEdit_AddComment_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
```



```

<wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/AddCommentResponse"
message="tns4:IPptEdit_AddComment_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptEdit_AddComment_InputMessage** request message, and the protocol server responds with an **IPptEdit_AddComment_OutputMessage** response message

3.1.4.1.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_AddComment_InputMessage	The request WSDL message for the AddComment WSDL operation.
IPptEdit_AddComment_OutputMessage	The response WSDL message for the AddComment WSDL operation.

3.1.4.1.1.1 IPptEdit_AddComment_InputMessage

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

The **SOAP action** value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/AddComment

```

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

3.1.4.1.1.2 IPptEdit_AddComment_OutputMessage

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

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

3.1.4.1.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
AddComment	The input data for the AddComment WSDL operation.
AddCommentResponse	The result data for the AddComment WSDL operation.

3.1.4.1.2.1 AddComment

The **AddComment** element specifies the input data for the **AddComment** WSDL operation.

```

<xs:element name="AddComment" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="text" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="timeZoneBias" type="xs:int"/>
      <xs:element minOccurs="0" name="parentAuthor" type="xs:int"/>
      <xs:element minOccurs="0" name="parentIndex" type="xs:int"/>
      <xs:element minOccurs="0" name="fHasPos" type="xs:boolean"/>
      <xs:element minOccurs="0" name="x" type="xs:int"/>
      <xs:element minOccurs="0" name="y" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the comment that is being modified.

text: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the contents of the comment.

timeZoneBias: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the difference between the local time of the user and UTC, in minutes.

parentAuthor: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the identifier of the author of another comment. If this comment is not a reply, the value **MUST** be -1.

parentIndex: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the identifier of another comment. If this comment is not a reply, the value **MUST** be -1.

fHasPos: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that specifies if the comment has a specific position on the presentation slide.

x: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the x coordinate of the comment.

y: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the y coordinate of the comment.

3.1.4.1.2.2 AddCommentResponse

The **AddCommentResponse** element specifies the result data for the **AddComment** WSDL operation.

```

<xs:element name="AddCommentResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="AddCommentResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

AddCommentResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

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 ApplyShapeFill

The **ApplyShapeFill** operation SHOULD [<71>](#) be used to modify the color of a shape in a presentation.

The following is the WSDL port type specification of the **ApplyShapeFill** WSDL operation.

```
<wsdl:operation name="ApplyShapeFill" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ApplyShapeFill"
    message="tns4:IPptEdit_ApplyShapeFill_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ApplyShapeFillResponse"
    message="tns4:IPptEdit_ApplyShapeFill_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ApplyShapeFill_InputMessage** request message, and the protocol server responds with an **IPptEdit_ApplyShapeFill_OutputMessage** response message.

3.1.4.2.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ApplyShapeFill_InputMessage	The request WSDL message for the ApplyShapeFill WSDL operation.
IPptEdit_ApplyShapeFill_OutputMessage	The response WSDL message for the ApplyShapeFill WSDL operation.

3.1.4.2.1.1 IPptEdit_ApplyShapeFill_InputMessage

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

The SOAP action value is:

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

3.1.4.2.1.2 IPptEdit_ApplyShapeFill_OutputMessage

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

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

3.1.4.2.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ApplyShapeFill	The input data for the ApplyShapeFill WSDL operation.
ApplyShapeFillResponse	The result data for the ApplyShapeFill WSDL operation.

3.1.4.2.2.1 ApplyShapeFill

The **ApplyShapeFill** element specifies the input data for the **ApplyShapeFill** WSDL operation.

```
<xs:element name="ApplyShapeFill" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="RGBColor" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="themeColor" type="xs:int"/>
      <xs:element minOccurs="0" name="colorLuminance" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="fTintColor" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section 3.1.4.27.

slideId: An **ST_SlideId** ([ISO/IEC29500-1:2011] section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape that is being modified.

RGBColor: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the red, green, and blue components of the color. This element MAY be present.

themeColor: An **int** ([XMLSCHEMA2] section 3.3.17) that specifies a preset theme color. This element is ignored if the **RGBColor** element is present.

colorLuminance: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the luminance of the color.

fTintColor: A **boolean** ([XMLSCHEMA2] section 3.2.2) that specifies if the color is a tint.

3.1.4.2.2 ApplyShapeFillResponse

The **ApplyShapeFillResponse** element specifies the result data for the **ApplyShapeFill** WSDL operation.

```
<xs:element name="ApplyShapeFillResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ApplyShapeFillResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ApplyShapeFillResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.2.3 Complex Types

None.

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 ApplyShapeOutlineColor

The **ApplyShapeOutlineColor** operation SHOULD [<72>](#) be used to modify the color of the outline of a shape in a presentation.

The following is the WSDL port type specification of the **ApplyShapeOutlineColor** WSDL operation.

```
<wsdl:operation name="ApplyShapeOutlineColor" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input>
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeOutlineColor"
    message="tns4:IPptEdit ApplyShapeOutlineColor InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output>
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeOutlineColorResponse"
    message="tns4:IPptEdit_ApplyShapeOutlineColor_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ApplyShapeOutlineColor_InputMessage** request message, and the protocol server responds with an **IPptEdit_ApplyShapeOutlineColor_OutputMessage** response message.

3.1.4.3.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ApplyShapeOutlineColor_InputMessage	The request WSDL message for the ApplyShapeOutlineColor WSDL operation.
IPptEdit_ApplyShapeOutlineColor_OutputMessage	The response WSDL message for the ApplyShapeOutlineColor WSDL operation.

3.1.4.3.1.1 IPptEdit_ApplyShapeOutlineColor_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ApplyShapeOutlineColor
```

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

3.1.4.3.1.2 IPptEdit_ApplyShapeOutlineColor_OutputMessage

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

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

3.1.4.3.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ApplyShapeOutlineColor	The input data for the ApplyShapeOutlineColor WSDL operation.
ApplyShapeOutlineColorResponse	The result data for the ApplyShapeOutlineColor WSDL operation.

3.1.4.3.2.1 ApplyShapeOutlineColor

The **ApplyShapeOutlineColor** element specifies the input data for the **ApplyShapeOutlineColor** WSDL operation.

```
<xs:element name="ApplyShapeOutlineColor" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```

    <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="RGBColor" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="themeColor" type="xs:int"/>
    <xs:element minOccurs="0" name="colorLuminance" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="fTintColor" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
</xs:element>

```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the shape that is being modified.

RGBColor: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the red, green, and blue components of the color. This element MAY be present.

themeColor: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies a preset theme color. This element is ignored if the RGBColor element is present.

colorLuminance: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the luminance of the color.

fTintColor: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that specifies if the color is a tint.

3.1.4.3.2.2 ApplyShapeOutlineColorResponse

The **ApplyShapeOutlineColorResponse** element specifies the result data for the **ApplyShapeOutlineColor** WSDL operation.

```

<xs:element name="ApplyShapeOutlineColorResponse"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
        name="ApplyShapeOutlineColorResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

ApplyShapeOutlineColorResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

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 ApplyShapeOutlineDashStyle

The **ApplyShapeOutlineDashStyle** operation SHOULD [<73>](#) be used to modify the dash type of the outline of a shape.

The following is the WSDL port type specification of the **ApplyShapeOutlineDashStyle** WSDL operation.

```
<wsdl:operation name="ApplyShapeOutlineDashStyle"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ApplyShapeOutlineDashStyle"
    message="tns4:IPptEdit_ApplyShapeOutlineDashStyle_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ApplyShapeOutlineDashStyleResponse"
    message="tns4:IPptEdit_ApplyShapeOutlineDashStyle_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ApplyShapeOutlineDashStyle_InputMessage** request message, and the protocol server responds with an **IPptEdit_ApplyShapeOutlineDashStyle_OutputMessage** response message.

3.1.4.4.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ApplyShapeOutlineDashStyle_InputMessage	The request WSDL message for the ApplyShapeOutlineDashStyle WSDL operation.
IPptEdit_ApplyShapeOutlineDashStyle_OutputMessage	The response WSDL message for the ApplyShapeOutlineDashStyle WSDL operation.

3.1.4.4.1.1 IPptEdit_ApplyShapeOutlineDashStyle_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ApplyShapeOutlineDashStyle
```


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

3.1.4.4.1.2 IPptEdit_ApplyShapeOutlineDashStyle_OutputMessage

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

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

3.1.4.4.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ApplyShapeOutlineDashStyle	The input data for the ApplyShapeOutlineDashStyle WSDL operation.
ApplyShapeOutlineDashStyleResponse	The result data for the ApplyShapeOutlineDashStyle WSDL operation.

3.1.4.4.2.1 ApplyShapeOutlineDashStyle

The **ApplyShapeOutlineDashStyle** element specifies the input data for the **ApplyShapeOutlineDashStyle** WSDL operation.

```
<xs:element name="ApplyShapeOutlineDashStyle" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="dashType" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape that is being modified.

dashType: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the dash type.

Value	Meaning
0x00	Solid.
0x01	Round dot.
0x02	Square dot.
0x03	Dash.
0x04	Dash dot.

Value	Meaning
0x05	Long dash.
0x06	Long dash dot.
0x07	Long dash dot dot.

3.1.4.4.2 ApplyShapeOutlineDashStyleResponse

The **ApplyShapeOutlineDashStyleResponse** element specifies the result data for the **ApplyShapeOutlineDashStyle** WSDL operation.

```
<xs:element name="ApplyShapeOutlineDashStyleResponse"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
        name="ApplyShapeOutlineDashStyleResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ApplyShapeOutlineDashStyleResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

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 ApplyShapeOutlineEndStyle

The **ApplyShapeOutlineEndStyle** operation SHOULD [<74>](#) be used to modify the end style of a line in a presentation.

The following is the WSDL port type specification of the **ApplyShapeOutlineEndStyle** WSDL operation.

```
<wsdl:operation name="ApplyShapeOutlineEndStyle"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
```

```

    <wsdl:input
      wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
      ServerInternalService/IPptEdit/ApplyShapeOutlineEndStyle"
      message="tns4:IPptEdit_ApplyShapeOutlineEndStyle_InputMessage"
      xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
    <wsdl:output
      wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
      ServerInternalService/IPptEdit/ApplyShapeOutlineEndStyleResponse"
      message="tns4:IPptEdit_ApplyShapeOutlineEndStyle_OutputMessage"
      xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  </wsdl:operation>

```

The protocol client sends an **IPptEdit_ApplyShapeOutlineEndStyle_InputMessage** request message, and the protocol server responds with an **IPptEdit_ApplyShapeOutlineEndStyle_OutputMessage** response message.

3.1.4.5.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ApplyShapeOutlineEndStyle_InputMessage	The request WSDL message for the ApplyShapeOutlineEndStyle WSDL operation.
IPptEdit_ApplyShapeOutlineEndStyle_OutputMessage	The response WSDL message for the ApplyShapeOutlineEndStyle WSDL operation.

3.1.4.5.1.1 IPptEdit_ApplyShapeOutlineEndStyle_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ApplyShapeOutlineEndStyle

```

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

3.1.4.5.1.2 IPptEdit_ApplyShapeOutlineEndStyle_OutputMessage

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

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

3.1.4.5.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ApplyShapeOutlineEndStyle	The input data for the ApplyShapeOutlineEndStyle WSDL operation.
ApplyShapeOutlineEndStyleResponse	The result data for the ApplyShapeOutlineEndStyle WSDL

Element	Description
	operation.

3.1.4.5.2.1 ApplyShapeOutlineEndStyle

The **ApplyShapeOutlineEndStyle** element specifies the input data for the **ApplyShapeOutlineEndStyle** WSDL operation.

```
<xs:element name="ApplyShapeOutlineEndStyle" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="headEndType" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="tailEndType" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the presentation containing the line that is being modified, as specified in section 3.1.4.27.

slideId: An **ST_SlideId** ([ISO/IEC29500-1:2011] section 19.7.13) that specifies the identifier of the presentation slide containing the line that is being modified.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the line that is being modified.

headEndType: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the end style for the head of the line.

Value	Meaning
0x00	None.
0x01	Triangle.
0x02	Arrow.
0x03	Stealth.
0x04	Diamond.
0x05	Oval.

tailEndType: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the end style for the tail of the line.

Value	Meaning
0x00	None.
0x01	Triangle.

Value	Meaning
0x02	Arrow.
0x03	Stealth.
0x04	Diamond.
0x05	Oval.

3.1.4.5.2.2 ApplyShapeOutlineEndStyleResponse

The **ApplyShapeOutlineEndStyleResponse** element specifies the result data for the **ApplyShapeOutlineEndStyle** WSDL operation.

```
<xs:element name="ApplyShapeOutlineEndStyleResponse"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
        name="ApplyShapeOutlineEndStyleResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ApplyShapeOutlineEndStyleResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

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 ApplyShapeOutlineWidth

The **ApplyShapeOutlineWidth** operation SHOULD [<75>](#) be used to modify the width of the outline of a shape.

The following is the WSDL port type specification of the **ApplyShapeOutlineWidth** WSDL operation.

```
<wsdl:operation name="ApplyShapeOutlineWidth" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
```

```

    <wsdl:input
      wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
      ServerInternalService/IPptEdit/ApplyShapeOutlineWidth"
      message="tns4:IPptEdit_ApplyShapeOutlineWidth_InputMessage"
      xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
    <wsdl:output
      wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
      ServerInternalService/IPptEdit/ApplyShapeOutlineWidthResponse"
      message="tns4:IPptEdit_ApplyShapeOutlineWidth_OutputMessage"
      xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  </wsdl:operation>

```

The protocol client sends an **IPptEdit_ApplyShapeOutlineWidth_InputMessage** request message, and the protocol server responds with an **IPptEdit_ApplyShapeOutlineWidth_OutputMessage** response message.

3.1.4.6.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ApplyShapeOutlineWidth_InputMessage	The request WSDL message for the ApplyShapeOutlineWidth WSDL operation.
IPptEdit_ApplyShapeOutlineWidth_OutputMessage	The response WSDL message for the ApplyShapeOutlineWidth WSDL operation.

3.1.4.6.1.1 IPptEdit_ApplyShapeOutlineWidth_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ApplyShapeOutlineWidth

```

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

3.1.4.6.1.2 IPptEdit_ApplyShapeOutlineWidth_OutputMessage

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

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

3.1.4.6.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ApplyShapeOutlineWidth	The input data for the ApplyShapeOutlineWidth WSDL operation.
ApplyShapeOutlineWidthResponse	The result data for the ApplyShapeOutlineWidth WSDL operation.

3.1.4.6.2.1 ApplyShapeOutlineWidth

The **ApplyShapeOutlineWidth** element specifies the input data for the **ApplyShapeOutlineWidth** WSDL operation.

```
<xs:element name="ApplyShapeOutlineWidth" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="width" type="xs:double"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the shape that is being modified.

width: A **double** ([\[XMLSCHEMA2\]](#) section 3.3.5) that specifies the width of the outline in points. The value MUST be greater than zero and less than or equal to 1584.

3.1.4.6.2.2 ApplyShapeOutlineWidthResponse

The **ApplyShapeOutlineWidthResponse** element specifies the result data for the **ApplyShapeOutlineWidth** WSDL operation.

```
<xs:element name="ApplyShapeOutlineWidthResponse"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
        name="ApplyShapeOutlineWidthResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ApplyShapeOutlineWidthResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

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 ApplyShapeStyle

The **ApplyShapeStyle** operation SHOULD [<76>](#) be used to apply a style to a shape in a presentation.

The following is the WSDL port type specification of the **ApplyShapeStyle** WSDL operation.

```
<wsdl:operation name="ApplyShapeStyle" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ApplyShapeStyle"
    message="tns4:IPptEdit_ApplyShapeStyle_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ApplyShapeStyleResponse"
    message="tns4:IPptEdit_ApplyShapeStyle_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ApplyShapeStyle_InputMessage** request message, and the protocol server responds with an **IPptEdit_ApplyShapeStyle_OutputMessage** response message.

3.1.4.7.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ApplyShapeStyle_InputMessage	The request WSDL message for the ApplyShapeStyle WSDL operation.
IPptEdit_ApplyShapeStyle_OutputMessage	The response WSDL message for the ApplyShapeStyle WSDL operation.

3.1.4.7.1.1 IPptEdit_ApplyShapeStyle_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ApplyShapeStyle
```

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

3.1.4.7.1.2 IPptEdit_ApplyShapeStyle_OutputMessage

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

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

3.1.4.7.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ApplyShapeStyle	The input data for the ApplyShapeStyle WSDL operation.
ApplyShapeStyleResponse	The result data for the ApplyShapeStyle WSDL operation.

3.1.4.7.2.1 ApplyShapeStyle

The **ApplyShapeStyle** element specifies the input data for the **ApplyShapeStyle** WSDL operation.

```
<xs:element name="ApplyShapeStyle" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="styleId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape that is being modified.

styleId: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the style. The value **MUST** be greater than or equal to zero and less than 42.

3.1.4.7.2.2 ApplyShapeStyleResponse

The **ApplyShapeStyleResponse** element specifies the result data for the **ApplyShapeStyle** WSDL operation.

```
<xs:element name="ApplyShapeStyleResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ApplyShapeStyleResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ApplyShapeStyleResult: A **ServiceResult** (section [2.2.4.43](#)) element that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)) [element](#).

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 ApplyTheme

The **ApplyTheme** operation SHOULD [<77>](#) be used to change the theme applied to a presentation.

The following is the WSDL port type specification of the **ApplyTheme** WSDL operation.

```
<wsdl:operation name="ApplyTheme" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ApplyTheme" message="tns4:IPptEdit_ApplyTheme_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ApplyThemeResponse"
    message="tns4:IPptEdit_ApplyTheme_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ApplyTheme_InputMessage** request message, and the protocol server responds with an **IPptEdit_ApplyTheme_OutputMessage** response message.

3.1.4.8.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ApplyTheme_InputMessage	The request WSDL message for the ApplyTheme WSDL operation.
IPptEdit_ApplyTheme_OutputMessage	The response WSDL message for the ApplyTheme WSDL operation.

3.1.4.8.1.1 IPptEdit_ApplyTheme_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ApplyTheme
```

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

3.1.4.8.1.2 IPptEdit_ApplyTheme_OutputMessage

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

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

3.1.4.8.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ApplyTheme	The input data for the ApplyTheme WSDL operation.
ApplyThemeResponse	The result data for the ApplyTheme WSDL operation.

3.1.4.8.2.1 ApplyTheme

The **ApplyTheme** element specifies the input data for the **ApplyTheme** WSDL operation.

```
<xs:element name="ApplyTheme" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="themeFamilyId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="variantIndex" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([ISO/IEC29500-1:2011] section 19.7.13) that specifies the identifier of the presentation slide.

themeFamilyId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the theme to be applied to the presentation.

variantIndex: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the theme variant to be applied to the presentation.

3.1.4.8.2.2 ApplyThemeResponse

The **ApplyThemeResponse** element specifies the result data for the **ApplyTheme** WSDL operation.

```
<xs:element name="ApplyThemeResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ApplyThemeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ApplyThemeResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

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 ArrangeShape

The **ArrangeShape** operation SHOULD [<78>](#) be used to rearrange a shape in a presentation.

The following is the WSDL port type specification of the **ArrangeShape** WSDL operation.

```
<wsdl:operation name="ArrangeShape" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ArrangeShape"
message="tns4:IPptEdit_ArrangeShape_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ArrangeShapeResponse"
message="tns4:IPptEdit_ArrangeShape_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ArrangeShape_InputMessage** request message, and the protocol server responds with an **IPptEdit_ArrangeShape_OutputMessage** response message.

3.1.4.9.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ArrangeShape_InputMessage	The request WSDL message for the ArrangeShape WSDL operation.
IPptEdit_ArrangeShape_OutputMessage	The response WSDL message for the ArrangeShape WSDL operation.

3.1.4.9.1.1 IPptEdit_ArrangeShape_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ArrangeShape
```

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

3.1.4.9.1.2 IPptEdit_ArrangeShape_OutputMessage

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

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

3.1.4.9.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ArrangeShape	The input data for the ArrangeShape WSDL operation.
ArrangeShapeResponse	The result data for the ArrangeShape WSDL operation.

3.1.4.9.2.1 ArrangeShape

The **ArrangeShape** element specifies the input data for the **ArrangeShape** WSDL operation.

```
<xs:element name="ArrangeShape" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="orderMode" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([ISO/IEC29500-1:2011] section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape that is being modified.

orderMode: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies how to rearrange the shape.

Value	Description
0x0	Bring the shape to the front, in front of all other shapes.
0x1	Bring the shape towards the front by one position.
0x2	Send the shape to the back, behind all other shapes.
0x3	Send the shape toward the back by one position.

3.1.4.9.2.2 ArrangeShapeResponse

The **ArrangeShapeResponse** element specifies the result data for the **ArrangeShape** WSDL operation.

```
<xs:element name="ArrangeShapeResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ArrangeShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ArrangeShapeResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

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.

3.1.4.10 ChangeLayout

The **ChangeLayout** operation is used to change the layout of a slide.

The following is the WSDL port type specification of the **ChangeLayout** WSDL operation.

```
<wsdl:operation name="ChangeLayout" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ChangeLayout"
    message="tns4:IPptEdit_ChangeLayout_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ChangeLayoutResponse"
    message="tns4:IPptEdit_ChangeLayout_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ChangeLayout_InputMessage** request message, and the protocol server responds with an **IPptEdit_ChangeLayout_OutputMessage** response message.

3.1.4.10.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ChangeLayout_InputMessage	The request WSDL message for the ChangeLayout WSDL operation.
IPptEdit_ChangeLayout_OutputMessage	The response WSDL message for the ChangeLayout WSDL operation.

3.1.4.10.1.1 IPptEdit_ChangeLayout_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ChangeLayout
```

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

3.1.4.10.1.2 IPptEdit_ChangeLayout_OutputMessage

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

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

3.1.4.10.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ChangeLayout	The input data for the ChangeLayout WSDL operation.
ChangeLayoutResponse	The result data for the ChangeLayout WSDL operation.

3.1.4.10.2.1 ChangeLayout

The **ChangeLayout** element specifies the input data for the **ChangeLayout** WSDL operation.

```
<xs:element name="ChangeLayout" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="mainMasterId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="contentMasterId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

mainMasterId: An **ST_SlideMasterId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.16) that specifies the identifier for the main master slide of the presentation slide.

contentMasterId: An **ST_SlideLayoutId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.14) that specifies the identifier for the slide layout of the presentation slide.

3.1.4.10.2.2 ChangeLayoutResponse

The **ChangeLayoutResponse** element specifies the result data for the **ChangeLayout** WSDL operation.

```
<xs:element name="ChangeLayoutResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ChangeLayoutResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ChangeLayoutResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.10.3 Complex Types

None.

3.1.4.10.4 Simple Types

None.

3.1.4.10.5 Attributes

None.

3.1.4.10.6 Groups

None.

3.1.4.10.7 Attribute Groups

None.

3.1.4.11 ChangePictureStyle

The **ChangePictureStyle** operation is used to change the style applied to a picture.

The following is the WSDL port type specification of the **ChangePictureStyle** WSDL operation.

```
<wsdl:operation name="ChangePictureStyle" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ChangePictureStyle"
    message="tns4:IPptEdit_ChangePictureStyle_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ChangePictureStyleResponse"
    message="tns4:IPptEdit_ChangePictureStyle_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ChangePictureStyle_InputMessage** request message, and the protocol server responds with an **IPptEdit_ChangePictureStyle_OutputMessage** response message.

3.1.4.11.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ChangePictureStyle_InputMessage	The request WSDL message for the ChangePictureStyle WSDL operation.
IPptEdit_ChangePictureStyle_OutputMessage	The response WSDL message for the ChangePictureStyle WSDL operation.

3.1.4.11.1.1 IPptEdit_ChangePictureStyle_InputMessage

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

The SOAP action value is:

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

3.1.4.11.1.2 IPptEdit_ChangePictureStyle_OutputMessage

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

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

3.1.4.11.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ChangePictureStyle	The input data for the ChangePictureStyle WSDL operation.
ChangePictureStyleResponse	The result data for the ChangePictureStyle WSDL operation.

3.1.4.11.2.1 ChangePictureStyle

The **ChangePictureStyle** element specifies the input data for the **ChangePictureStyle** WSDL operation.

```
<xs:element name="ChangePictureStyle" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="pictureStyleId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([ISO/IEC29500-1:2011] section 19.7.13) that specifies the identifier of the presentation slide.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the picture.

pictureStyleId: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies which style the picture is changed to. The value **MUST** be greater than or equal to 15505 and less than or equal to 15532.

3.1.4.11.2.2 ChangePictureStyleResponse

The **ChangePictureStyleResponse** element specifies the result data for the **ChangePictureStyle** WSDL operation.

```

<xs:element name="ChangePictureStyleResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ChangePictureStyleResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

ChangePictureStyleResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.11.3 Complex Types

None.

3.1.4.11.4 Simple Types

None.

3.1.4.11.5 Attributes

None.

3.1.4.11.6 Groups

None.

3.1.4.11.7 Attribute Groups

None.

3.1.4.12 ChangeSmartArtColor

The **ChangeSmartArtColor** operation is used to change the color transform applied to a SmartArt diagram.

The following is the WSDL port type specification of the **ChangeSmartArtColor** WSDL operation.

```

<wsdl:operation name="ChangeSmartArtColor" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ChangeSmartArtColor"
message="tns4:IPptEdit_ChangeSmartArtColor_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ChangeSmartArtColorResponse"
message="tns4:IPptEdit_ChangeSmartArtColor_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptEdit_ChangeSmartArtColor_InputMessage** request message, and the protocol server responds with an **IPptEdit_ChangeSmartArtColor_OutputMessage** response message.

3.1.4.12.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ChangeSmartArtColor_InputMessage	The request WSDL message for the ChangeSmartArtColor WSDL operation.
IPptEdit_ChangeSmartArtColor_OutputMessage	The response WSDL message for the ChangeSmartArtColor WSDL operation.

3.1.4.12.1.1 IPptEdit_ChangeSmartArtColor_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ChangeSmartArtColor
```

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

3.1.4.12.1.2 IPptEdit_ChangeSmartArtColor_OutputMessage

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

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

3.1.4.12.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ChangeSmartArtColor	The input data for the ChangeSmartArtColor WSDL operation.
ChangeSmartArtColorResponse	The result data for the ChangeSmartArtColor WSDL operation.

3.1.4.12.2.1 ChangeSmartArtColor

The **ChangeSmartArtColor** element specifies the input data for the **ChangeSmartArtColor** WSDL operation.

```
<xs:element name="ChangeSmartArtColor" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="colorTransUri" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="category" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
```

</xs:element>

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation where the color transform of the SmartArt diagram is changed, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide where the color transform of the SmartArt diagram is changed.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the SmartArt diagram the color of which is changed.

colorTransUri: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the color transform which is applied to the SmartArt diagram.

category: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the category to which the color transform belongs.

3.1.4.12.2 ChangeSmartArtColorResponse

The **ChangeSmartArtColorResponse** element specifies the result data for the **ChangeSmartArtColor** WSDL operation.

```
<xs:element name="ChangeSmartArtColorResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ChangeSmartArtColorResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ChangeSmartArtColorResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.12.3 Complex Types

None.

3.1.4.12.4 Simple Types

None.

3.1.4.12.5 Attributes

None.

3.1.4.12.6 Groups

None.

3.1.4.12.7 Attribute Groups

None.

3.1.4.13 ChangeSmartArtLayout

The **ChangeSmartArtLayout** operation is used to change the layout applied to a SmartArt diagram.

The following is the WSDL port type specification of the **ChangeSmartArtLayout** WSDL operation.

```
<wsdl:operation name="ChangeSmartArtLayout" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ChangeSmartArtLayout"
    message="tns4:IPptEdit_ChangeSmartArtLayout_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ChangeSmartArtLayoutResponse"
    message="tns4:IPptEdit_ChangeSmartArtLayout_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ChangeSmartArtLayout_InputMessage** request message, and the protocol server responds with an **IPptEdit_ChangeSmartArtLayout_OutputMessage** response message.

3.1.4.13.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ChangeSmartArtLayout_InputMessage	The request WSDL message for the ChangeSmartArtLayout WSDL operation.
IPptEdit_ChangeSmartArtLayout_OutputMessage	The response WSDL message for the ChangeSmartArtLayout WSDL operation.

3.1.4.13.1.1 IPptEdit_ChangeSmartArtLayout_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ChangeSmartArtLayout
```

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

3.1.4.13.1.2 IPptEdit_ChangeSmartArtLayout_OutputMessage

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

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

3.1.4.13.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ChangeSmartArtLayout	The input data for the ChangeSmartArtLayout WSDL operation.
ChangeSmartArtLayoutResponse	The result data for the ChangeSmartArtLayout WSDL operation.

3.1.4.13.2.1 ChangeSmartArtLayout

The **ChangeSmartArtLayout** element specifies the input data for the **ChangeSmartArtLayout** WSDL operation.

```
<xs:element name="ChangeSmartArtLayout" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="layoutUri" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="category" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation where the layout of the SmartArt diagram is changed, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide where the layout of the SmartArt diagram is changed.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the SmartArt diagram the layout of which is changed.

layoutUri: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the layout which is applied to the SmartArt diagram.

category: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the category to which the layout belongs.

3.1.4.13.2.2 ChangeSmartArtLayoutResponse

The **ChangeSmartArtLayoutResponse** element specifies the result data for the **ChangeSmartArtLayout** WSDL operation.

```
<xs:element name="ChangeSmartArtLayoutResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ChangeSmartArtLayoutResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ChangeSmartArtLayoutResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.13.3 Complex Types

None.

3.1.4.13.4 Simple Types

None.

3.1.4.13.5 Attributes

None.

3.1.4.13.6 Groups

None.

3.1.4.13.7 Attribute Groups

None.

3.1.4.14 ChangeSmartArtStyle

The **ChangeSmartArtStyle** operation is used to change the style applied to a SmartArt diagram.

The following is the WSDL port type specification of the **ChangeSmartArtStyle** WSDL operation.

```
<wsdl:operation name="ChangeSmartArtStyle" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ChangeSmartArtStyle"
    message="tns4:IPptEdit_ChangeSmartArtStyle_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ChangeSmartArtStyleResponse"
    message="tns4:IPptEdit_ChangeSmartArtStyle_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ChangeSmartArtStyle_InputMessage** request message, and the protocol server responds with an **IPptEdit_ChangeSmartArtStyle_OutputMessage** response message.

3.1.4.14.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ChangeSmartArtStyle_InputMessage	The request WSDL message for the ChangeSmartArtStyle WSDL operation.
IPptEdit_ChangeSmartArtStyle_OutputMessage	The response WSDL message for the ChangeSmartArtStyle WSDL operation.

3.1.4.14.1.1 IPptEdit_ChangeSmartArtStyle_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ChangeSmartArtStyle
```

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

3.1.4.14.1.2 IPptEdit_ChangeSmartArtStyle_OutputMessage

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

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

3.1.4.14.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ChangeSmartArtStyle	The input data for the ChangeSmartArtStyle WSDL operation.
ChangeSmartArtStyleResponse	The result data for the ChangeSmartArtStyle WSDL operation.

3.1.4.14.2.1 ChangeSmartArtStyle

The **ChangeSmartArtStyle** element specifies the input data for the **ChangeSmartArtStyle** WSDL operation.

```
<xs:element name="ChangeSmartArtStyle" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="styleUri" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="category" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the presentation where the style of the SmartArt diagram is changed, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([ISO/IEC29500-1:2011] section 19.7.13) that specifies the identifier of the presentation slide where the style of the SmartArt diagram is changed.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the SmartArt diagram the style of which is changed.

styleUri: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the style which is applied to the SmartArt diagram.

category: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the category to which the style belongs.

3.1.4.14.2 ChangeSmartArtStyleResponse

The **ChangeSmartArtStyleResponse** element specifies the result data for the **ChangeSmartArtStyle** WSDL operation.

```
<xs:element name="ChangeSmartArtStyleResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ChangeSmartArtStyleResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ChangeSmartArtStyleResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.14.3 Complex Types

None.

3.1.4.14.4 Simple Types

None.

3.1.4.14.5 Attributes

None.

3.1.4.14.6 Groups

None.

3.1.4.14.7 Attribute Groups

None.

3.1.4.15 ClearPlaceholder

The **ClearPlaceholder** is used to delete a shape in a presentation.

The following is the WSDL port type specification of the **ClearPlaceholder** WSDL operation.

```
<wsdl:operation name="ClearPlaceholder" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ClearPlaceholder"
message="tns4:IPptEdit_ClearPlaceholder_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ClearPlaceholderResponse"
message="tns4:IPptEdit_ClearPlaceholder_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ClearPlaceholder_InputMessage** request message, and the protocol server responds with an **IPptEdit_ClearPlaceholder_OutputMessage** response message.

3.1.4.15.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ClearPlaceholder_InputMessage	The request WSDL message for the ClearPlaceholder WSDL operation.
IPptEdit_ClearPlaceholder_OutputMessage	The response WSDL message for the ClearPlaceholder WSDL operation.

3.1.4.15.1.1 IPptEdit_ClearPlaceholder_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ClearPlaceholder
```

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

3.1.4.15.1.2 IPptEdit_ClearPlaceholder_OutputMessage

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

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

3.1.4.15.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ClearPlaceholder	The input data for the ClearPlaceholder WSDL operation.
ClearPlaceholderResponse	The result data for the ClearPlaceholder WSDL operation.

3.1.4.15.2.1 ClearPlaceholder

The **ClearPlaceholder** element specifies the input data for the **ClearPlaceholder** WSDL operation.

```
<xs:element name="ClearPlaceholder" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is deleted, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is deleted.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape that is deleted.

3.1.4.15.2.2 ClearPlaceholderResponse

The **ClearPlaceholderResponse** element specifies the result data for the **ClearPlaceholder** WSDL operation.

```
<xs:element name="ClearPlaceholderResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ClearPlaceholderResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ClearPlaceholderResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.15.3 Complex Types

None.

3.1.4.15.4 Simple Types

None.

3.1.4.15.5 Attributes

None.

3.1.4.15.6 Groups

None.

3.1.4.15.7 Attribute Groups

None.

3.1.4.16 DeleteComment

The **DeleteComment** operation SHOULD [<79>](#) be used to delete a comment from a presentation.

The following is the WSDL port type specification of the **DeleteComment** WSDL operation.

```
<wsdl:operation name="DeleteComment" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/DeleteComment"
message="tns4:IPptEdit_DeleteComment_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
```

```

<wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/DeleteCommentResponse"
message="tns4:IPptEdit_DeleteComment_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptEdit_DeleteComment_InputMessage** request message, and the protocol server responds with an **IPptEdit_DeleteComment_OutputMessage** response message

3.1.4.16.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_DeleteComment_InputMessage	The request WSDL message for the DeleteComment WSDL operation.
IPptEdit_DeleteComment_OutputMessage	The response WSDL message for the DeleteComment WSDL operation.

3.1.4.16.1.1 IPptEdit_DeleteComment_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/DeleteComment

```

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

3.1.4.16.1.2 IPptEdit_DeleteComment_OutputMessage

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

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

3.1.4.16.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
DeleteComment	The input data for the DeleteComment WSDL operation.
DeleteCommentResponse	The result data for the DeleteComment WSDL operation.

3.1.4.16.2.1 DeleteComment

The **DeleteComment** element specifies the input data for the **DeleteComment** WSDL operation.

```

<xs:element name="DeleteComment" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="authorId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="index" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the comment that is being deleted.

authorId: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the identifier of the author of the comment.

index: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the identifier of the comment within its author's comments.

3.1.4.16.2 DeleteCommentResponse

The **DeleteCommentResponse** element specifies the result data for the **DeleteComment** WSDL operation.

```

<xs:element name="DeleteCommentResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="DeleteCommentResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

DeleteCommentResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.16.3 Complex Types

None.

3.1.4.16.4 Simple Types

None.

3.1.4.16.5 Attributes

None.

3.1.4.16.6 Groups

None.

3.1.4.16.7 Attribute Groups

None.

3.1.4.17 DeleteSlide

The **DeleteSlide** operation is used to delete a presentation slide from a presentation.

The following is the WSDL port type specification of the **DeleteSlide** WSDL operation.

```
<wsdl:operation name="DeleteSlide" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/DeleteSlide" message="tns4:IPptEdit DeleteSlide InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/DeleteSlideResponse"
    message="tns4:IPptEdit DeleteSlide OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_DeleteSlide_InputMessage** request message, and the protocol server responds with an **IPptEdit_DeleteSlide_OutputMessage** response message.

3.1.4.17.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_DeleteSlide_InputMessage	The request WSDL message for the DeleteSlide WSDL operation.
IPptEdit_DeleteSlide_OutputMessage	The response WSDL message for the DeleteSlide WSDL operation.

3.1.4.17.1.1 IPptEdit_DeleteSlide_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/DeleteSlide
```

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

3.1.4.17.1.2 IPptEdit_DeleteSlide_OutputMessage

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

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

3.1.4.17.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
DeleteSlide	The input data for the DeleteSlide WSDL operation.
DeleteSlideResponse	The result data for the DeleteSlide WSDL operation.

3.1.4.17.2.1 DeleteSlide

The **DeleteSlide** element specifies the input data for the **DeleteSlide** WSDL operation.

```
<xs:element name="DeleteSlide" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide to delete.

3.1.4.17.2.2 DeleteSlideResponse

The **DeleteSlideResponse** element specifies the result data for the **DeleteSlide** WSDL operation.

```
<xs:element name="DeleteSlideResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="DeleteSlideResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

DeleteSlideResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.17.3 Complex Types

None.

3.1.4.17.4 Simple Types

None.

3.1.4.17.5 Attributes

None.

3.1.4.17.6 Groups

None.

3.1.4.17.7 Attribute Groups

None.

3.1.4.18 DuplicateShape

The **DuplicateShape** operation SHOULD [<80>](#) be used to duplicate a shape in a presentation.

The following is the WSDL port type specification of the **DuplicateShape** WSDL operation.

```
<wsdl:operation name="DuplicateShape" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/DuplicateShape"
    message="tns4:IPptEdit DuplicateShape InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/DuplicateShapeResponse"
    message="tns4:IPptEdit_DuplicateShape_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_DuplicateShape_InputMessage** request message, and the protocol server responds with an **IPptEdit_DuplicateShape_OutputMessage** response message.

3.1.4.18.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_DuplicateShape_InputMessage	The request WSDL message for the DuplicateShape WSDL operation.
IPptEdit_DuplicateShape_OutputMessage	The response WSDL message for the DuplicateShape WSDL operation.

3.1.4.18.1.1 IPptEdit_DuplicateShape_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/DuplicateShape
```

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

3.1.4.18.1.2 IPptEdit_DuplicateShape_OutputMessage

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

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

3.1.4.18.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
DuplicateShape	The input data for the DuplicateShape WSDL operation.
DuplicateShapeResponse	The result data for the DuplicateShape WSDL operation.

3.1.4.18.2.1 DuplicateShape

The **DuplicateShape** element specifies the input data for the **DuplicateShape** WSDL operation.

```
<xs:element name="DuplicateShape" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="sourceSlideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="sourceShapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="destinationSlideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="cloneShapePosX" type="xs:int"/>
      <xs:element minOccurs="0" name="cloneShapePosY" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is duplicated, as specified in section [3.1.4.27](#).

sourceSlideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that the formatting is copied from.

sourceShapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the shape that the formatting is copied from. The shape **MUST NOT** be a group, SmartArt, table, chart, or ActiveX shape.

destinationSlideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that the formatting is applied to.

cloneShapePosX: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the x coordinate of the shape that is inserted.

cloneShapePosY: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the y coordinate of the shape that is inserted.

3.1.4.18.2.2 DuplicateShapeResponse

The **DuplicateShapeResponse** element specifies the result data for the **DuplicateShape** WSDL operation.

```
<xs:element name="DuplicateShapeResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
        name="DuplicateShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
```

</xs:element>

DuplicateShapeResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.18.3 Complex Types

None.

3.1.4.18.4 Simple Types

None.

3.1.4.18.5 Attributes

None.

3.1.4.18.6 Groups

None.

3.1.4.18.7 Attribute Groups

None.

3.1.4.19 DuplicateSlide

The **DuplicateSlide** operation is used to duplicate a presentation slide.

The following is the WSDL port type specification of the **DuplicateSlide** WSDL operation.

```
<wsdl:operation name="DuplicateSlide" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/DuplicateSlide"
    message="tns4:IPptEdit_DuplicateSlide_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/DuplicateSlideResponse"
    message="tns4:IPptEdit_DuplicateSlide_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_DuplicateSlide_InputMessage** request message, and the protocol server responds with an **IPptEdit_DuplicateSlide_OutputMessage** response message.

3.1.4.19.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_DuplicateSlide_InputMessage	The request WSDL message for the DuplicateSlide WSDL operation.

Message	Description
IPptEdit_DuplicateSlide_OutputMessage	The response WSDL message for the DuplicateSlide WSDL operation.

3.1.4.19.1.1 IPptEdit_DuplicateSlide_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/DuplicateSlide
```

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

3.1.4.19.1.2 IPptEdit_DuplicateSlide_OutputMessage

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

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

3.1.4.19.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
DuplicateSlide	The input data for the DuplicateSlide WSDL operation.
DuplicateSlideResponse	The result data for the DuplicateSlide WSDL operation.

3.1.4.19.2.1 DuplicateSlide

The **DuplicateSlide** element specifies the input data for the **DuplicateSlide** WSDL operation.

```
<xs:element name="DuplicateSlide" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="srcId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="dstId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

srcId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide to duplicate.

dstId: A **ST_SlideId** ([ISO/IEC29500-1:2011] section 19.7.13) that specifies the identifier of the presentation slide that precedes the location of the duplicated slide.

3.1.4.19.2.2 DuplicateSlideResponse

The **DuplicateSlideResponse** element specifies the result data for the **DuplicateSlide** WSDL operation.

```
<xs:element name="DuplicateSlideResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="DuplicateSlideResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

DuplicateSlideResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.19.3 Complex Types

None.

3.1.4.19.4 Simple Types

None.

3.1.4.19.5 Attributes

None.

3.1.4.19.6 Groups

None.

3.1.4.19.7 Attribute Groups

None.

3.1.4.20 EditComment

The **EditComment** operation SHOULD [<81>](#) be used to change the contents of a comment in a presentation

The following is the WSDL port type specification of the **EditComment** WSDL operation.

```
<wsdl:operation name="EditComment" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/EditComment" message="tns4:IPptEdit_EditComment_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/EditCommentResponse"
message="tns4:IPptEdit_EditComment_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
```

```
</wsdl:operation>
```

The protocol client sends an **IPptEdit_EditComment_InputMessage** request message, and the protocol server responds with an **IPptEdit_EditComment_OutputMessage** response message.

3.1.4.20.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_EditComment_InputMessage	The request WSDL message for the EditComment WSDL operation.
IPptEdit_EditComment_OutputMessage	The response WSDL message for the EditComment WSDL operation.

3.1.4.20.1.1 IPptEdit_EditComment_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/EditComment
```

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

3.1.4.20.1.2 IPptEdit_EditComment_OutputMessage

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

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

3.1.4.20.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
EditComment	The input data for the EditComment WSDL operation.
EditCommentResponse	The result data for the EditComment WSDL operation.

3.1.4.20.2.1 EditComment

The **EditComment** element specifies the input data for the **EditComment** WSDL operation.

```
<xs:element name="EditComment" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>
```

```

    <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="text" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="authorId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="index" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
</xs:element>

```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the comment that is being modified.

text: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the new contents of the comment.

authorId: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the identifier of the author of the comment.

index: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the identifier of the comment within its author's comments.

3.1.4.20.2 EditCommentResponse

The **EditCommentResponse** element specifies the result data for the **EditComment** WSDL operation.

```

<xs:element name="EditCommentResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="EditCommentResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

EditCommentResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.20.3 Complex Types

None.

3.1.4.20.4 Simple Types

None.

3.1.4.20.5 Attributes

None.

3.1.4.20.6 Groups

None.

3.1.4.20.7 Attribute Groups

None.

3.1.4.21 FlipShape

The **FlipShape** operation SHOULD [<82>](#) be used to flip a shape vertically or horizontally in a presentation.

The following is the WSDL port type specification of the **FlipShape** WSDL operation.

```
<wsdl:operation name="FlipShape" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/FlipShape" message="tns4:IPptEdit_FlipShape_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/FlipShapeResponse"
    message="tns4:IPptEdit_FlipShape_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_FlipShape_InputMessage** request message, and the protocol server responds with an **IPptEdit_FlipShape_OutputMessage** response message.

3.1.4.21.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_FlipShape_InputMessage	The request WSDL message for the FlipShape WSDL operation.
IPptEdit_FlipShape_OutputMessage	The response WSDL message for the FlipShape WSDL operation.

3.1.4.21.1.1 IPptEdit_FlipShape_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/FlipShape
```

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

3.1.4.21.1.2 IPptEdit_FlipShape_OutputMessage

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

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

3.1.4.21.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
FlipShape	The input data for the FlipShape WSDL operation.
FlipShapeResponse	The result data for the FlipShape WSDL operation.

3.1.4.21.2.1 FlipShape

The **FlipShape** element specifies the input data for the **FlipShape** WSDL operation.

```
<xs:element name="FlipShape" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="fHorizontal" type="xs:boolean"/>
      <xs:element minOccurs="0" name="fVertical" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the shape that is being modified.

fHorizontal: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that specifies whether the shape will be flipped horizontally.

fVertical: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that specifies whether the shape will be flipped vertically.

3.1.4.21.2.2 FlipShapeResponse

The **FlipShapeResponse** element specifies the result data for the **FlipShape** WSDL operation.

```
<xs:element name="FlipShapeResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="FlipShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

FlipShapeResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.21.3 Complex Types

None.

3.1.4.21.4 Simple Types

None.

3.1.4.21.5 Attributes

None.

3.1.4.21.6 Groups

None.

3.1.4.21.7 Attribute Groups

None.

3.1.4.22 GetCoauthUpdates

The **GetCoauthUpdates** operation SHOULD [<83>](#) be used to retrieve co-authoring state for a presentation.

The following is the WSDL port type specification of the **GetCoauthUpdates** WSDL operation.

```
<wsdl:operation name="GetCoauthUpdates" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/GetCoauthUpdates"
    message="tns4:IPptEdit_GetCoauthUpdates_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/GetCoauthUpdatesResponse"
    message="tns4:IPptEdit_GetCoauthUpdates_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_GetCoauthUpdates_InputMessage** request message, and the protocol server responds with an **IPptEdit_GetCoauthUpdates_OutputMessage** response message.

3.1.4.22.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_GetCoauthUpdates_InputMessage	The request WSDL message for the GetCoauthUpdates WSDL operation.
IPptEdit_GetCoauthUpdates_OutputMessage	The response WSDL message for the GetCoauthUpdates WSDL operation.

3.1.4.22.1.1 IPptEdit_GetCoauthUpdates_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/GetCoauthUpdates
```

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

3.1.4.22.1.2 IPptEdit_GetCoauthUpdates_OutputMessage

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

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

3.1.4.22.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetCoauthUpdates	The input data for the GetCoauthUpdates WSDL operation.
GetCoauthUpdatesResponse	The result data for the GetCoauthUpdates WSDL operation.

3.1.4.22.2.1 GetCoauthUpdates

The **GetCoauthUpdates** element specifies the input data for the **GetCoauthUpdates** WSDL operation.

```
<xs:element name="GetCoauthUpdates" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>  
      <xs:element minOccurs="0" name="lastToken" nillable="true" type="xs:string"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

presentationId: A string ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

lastToken: Reserved. MUST be ignored.

3.1.4.22.2.2 GetCoauthUpdatesResponse

The **GetCoauthUpdatesResponse** element specifies the result data for the **GetCoauthUpdates** WSDL operation.

```
<xs:element name="GetCoauthUpdatesResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"  
name="GetCoauthUpdatesResult" nillable="true" type="tns2:ServiceResult"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

GetCoauthUpdatesResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be a **CoauthUpdate** (section [2.2.4.18](#)).

3.1.4.22.3 Complex Types

None.

3.1.4.22.4 Simple Types

None.

3.1.4.22.5 Attributes

None.

3.1.4.22.6 Groups

None.

3.1.4.22.7 Attribute Groups

None.

3.1.4.23 GetEditPresInfo

The **GetEditPresInfo** operation is used to retrieve information about a presentation.

The following is the WSDL port type specification of the **GetEditPresInfo** WSDL operation.

```
<wsdl:operation name="GetEditPresInfo" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/GetEditPresInfo"
    message="tns4:IPptEdit_GetEditPresInfo_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/GetEditPresInfoResponse"
    message="tns4:IPptEdit_GetEditPresInfo_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_GetEditPresInfo_InputMessage** request message, and the protocol server responds with an **IPptEdit_GetEditPresInfo_OutputMessage** response message.

3.1.4.23.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_GetEditPresInfo_InputMessage	The request WSDL message for the GetEditPresInfo WSDL operation.
IPptEdit_GetEditPresInfo_OutputMessage	The response WSDL message for the GetEditPresInfo WSDL operation.

3.1.4.23.1.1 IPptEdit_GetEditPresInfo_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/GetEditPresInfo
```

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

3.1.4.23.1.2 IPptEdit_GetEditPresInfo_OutputMessage

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

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

3.1.4.23.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetEditPresInfo	The input data for the GetEditPresInfo WSDL operation.
GetEditPresInfoResponse	The result data for the GetEditPresInfo WSDL operation.

3.1.4.23.2.1 GetEditPresInfo

The **GetEditPresInfo** element specifies the input data for the **GetEditPresInfo** WSDL operation.

```
<xs:element name="GetEditPresInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="firstSlideIndex" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="numSlides" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="includedEditSlideIndex" type="xs:int"/>
      <xs:element minOccurs="0" name="clientWidth" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="clientHeight" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

firstSlideIndex: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the index of the first presentation slide for which to retrieve information.

numSlides: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the count of presentation slides for which to retrieve information.

includedEditSlideIndex: Reserved. MUST be ignored.

clientWidth: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the width of the client area.

clientHeight: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the height of the client area.

3.1.4.23.2 GetEditPresInfoResponse

The **GetEditPresInfoResponse** element specifies the result data for the **GetEditPresInfo** WSDL operation.

```
<xs:element name="GetEditPresInfoResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetEditPresInfoResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetEditPresInfoResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditPresentationInfo** (section [2.2.4.26](#)).

3.1.4.23.3 Complex Types

None.

3.1.4.23.4 Simple Types

None.

3.1.4.23.5 Attributes

None.

3.1.4.23.6 Groups

None.

3.1.4.23.7 Attribute Groups

None.

3.1.4.24 GetEditPresInfoGetEditSlideById

The **GetEditPresInfoGetEditSlideById** operation SHOULD [<84>](#) be used to retrieve information about a presentation.

The following is the WSDL port type specification of the **GetEditPresInfoGetEditSlideById** WSDL operation.

```
<wsdl:operation name="GetEditPresInfoGetEditSlideById"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/GetEditPresInfoGetEditSlideById"
message="tns4:IPptEdit_GetEditPresInfoGetEditSlideById_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
```

```

<wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/GetEditPresInfoGetEditSlideByIdResponse"
message="tns4:IPptEdit_GetEditPresInfoGetEditSlideById_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptEdit_GetEditPresInfoGetEditSlideById_InputMessage** request message, and the protocol server responds with an **IPptEdit_GetEditPresInfoGetEditSlideById_OutputMessage** response message.

3.1.4.24.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_GetEditPresInfoGetEditSlideById_InputMessage	The request WSDL message for the GetEditPresInfoGetEditSlideById WSDL operation.
IPptEdit_GetEditPresInfoGetEditSlideById_OutputMessage	The response WSDL message for the GetEditPresInfoGetEditSlideById WSDL operation.

3.1.4.24.1.1 IPptEdit_GetEditPresInfoGetEditSlideById_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/GetEditPresInfoGetEditSlideById

```

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

3.1.4.24.1.2 IPptEdit_GetEditPresInfoGetEditSlideById_OutputMessage

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

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

3.1.4.24.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetEditPresInfoGetEditSlideById	The input data for the GetEditPresInfoGetEditSlideById WSDL operation.
GetEditPresInfoGetEditSlideByIdResponse	The result data for the GetEditPresInfoGetEditSlideById WSDL operation.

3.1.4.24.2.1 GetEditPresInfoGetEditSlideById

The **GetEditPresInfoGetEditSlideById** element specifies the input data for the **GetEditPresInfoGetEditSlideById** WSDL operation.

```
<xs:element name="GetEditPresInfoGetEditSlideById"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="firstSlideIndex" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="numSlides" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="includedEditSlideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="clientWidth" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="clientHeight" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

firstSlideIndex: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the index of the first presentation slide for which to retrieve information.

numSlides: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the count of presentation slides for which to retrieve information.

includedEditSlideId: Reserved. MUST be ignored.

clientWidth: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the width of the client area.

clientHeight: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the height of the client area.

3.1.4.24.2.2 GetEditPresInfoGetEditSlideByIdResponse

The **GetEditPresInfoGetEditSlideByIdResponse** element specifies the result data for the **GetEditPresInfoGetEditSlideById** WSDL operation.

```
<xs:element name="GetEditPresInfoGetEditSlideByIdResponse"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetEditPresInfoGetEditSlideByIdResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetEditPresInfoGetEditSlideByIdResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditPresentationInfo** (section [2.2.4.26](#)).

3.1.4.24.3 Complex Types

None.

3.1.4.24.4 Simple Types

None.

3.1.4.24.5 Attributes

None.

3.1.4.24.6 Groups

None.

3.1.4.24.7 Attribute Groups

None.

3.1.4.25 GetEditPresResources

Reserved. MUST not be called.

The following is the WSDL port type specification of the **GetEditPresResources** WSDL operation.

```
<wsdl:operation name="GetEditPresResources" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/GetEditPresResources"
    message="tns4:IPptEdit_GetEditPresResources_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/GetEditPresResourcesResponse"
    message="tns4:IPptEdit_GetEditPresResources_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

Reserved. MUST not be called.

3.1.4.25.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_GetEditPresResources_InputMessage	The request WSDL message for the GetEditPresResources WSDL operation.
IPptEdit_GetEditPresResources_OutputMessage	The response WSDL message for the GetEditPresResources WSDL operation.

3.1.4.25.1.1 IPptEdit_GetEditPresResources_InputMessage

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

The SOAP action value is:

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

3.1.4.25.1.2 IPptEdit_GetEditPresResources_OutputMessage

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

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

3.1.4.25.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetEditPresResources	The input data for the GetEditPresResources WSDL operation.
GetEditPresResourcesResponse	The result data for the GetEditPresResources WSDL operation.

3.1.4.25.2.1 GetEditPresResources

The **GetEditPresResources** element specifies the input data for the **GetEditPresResources** WSDL operation.

```
<xs:element name="GetEditPresResources" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: Reserved. MUST be ignored.

3.1.4.25.2.2 GetEditPresResourcesResponse

The **GetEditPresResourcesResponse** element specifies the result data for the **GetEditPresResources** WSDL operation.

```
<xs:element name="GetEditPresResourcesResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetEditPresResourcesResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetEditPresResourcesResult: Reserved. MUST be ignored.

3.1.4.25.3 Complex Types

None.

3.1.4.25.4 Simple Types

None.

3.1.4.25.5 Attributes

None.

3.1.4.25.6 Groups

None.

3.1.4.25.7 Attribute Groups

None.

3.1.4.26 GetEditSlide

The **GetEditSlide** operation is used to retrieve information about a presentation slide.

The following is the WSDL port type specification of the **GetEditSlide** WSDL operation.

```
<wsdl:operation name="GetEditSlide" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/GetEditSlide"
    message="tns4:IPptEdit_GetEditSlide_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/GetEditSlideResponse"
    message="tns4:IPptEdit_GetEditSlide_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_GetEditSlide_InputMessage** request message, and the protocol server responds with an **IPptEdit_GetEditSlide_OutputMessage** response message.

3.1.4.26.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_GetEditSlide_InputMessage	The request WSDL message for the GetEditSlide WSDL operation.
IPptEdit_GetEditSlide_OutputMessage	The response WSDL message for the GetEditSlide WSDL operation.

3.1.4.26.1.1 IPptEdit_GetEditSlide_InputMessage

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

The SOAP action value is:

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/GetEditSlide

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

3.1.4.26.1.2 IPptEdit_GetEditSlide_OutputMessage

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

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

3.1.4.26.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetEditSlide	The input data for the GetEditSlide WSDL operation.
GetEditSlideResponse	The result data for the GetEditSlide WSDL operation.

3.1.4.26.2.1 GetEditSlide

The **GetEditSlide** element specifies the input data for the **GetEditSlide** WSDL operation.

```
<xs:element name="GetEditSlide" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide.

3.1.4.26.2.2 GetEditSlideResponse

The **GetEditSlideResponse** element specifies the result data for the **GetEditSlide** WSDL operation.

```
<xs:element name="GetEditSlideResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetEditSlideResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetEditSlideResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditSlide** (section [2.2.4.29](#)).

3.1.4.26.3 Complex Types

None.

3.1.4.26.4 Simple Types

None.

3.1.4.26.5 Attributes

None.

3.1.4.26.6 Groups

None.

3.1.4.26.7 Attribute Groups

None.

3.1.4.27 GetPresentationId

The **GetPresentationId** operation is used to retrieve an identifier for a presentation.

The following is the WSDL port type specification of the **GetPresentationId** WSDL operation.

```
<wsdl:operation name="GetPresentationId" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/GetPresentationId"
    message="tns4:IPptEdit_GetPresentationId_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/GetPresentationIdResponse"
    message="tns4:IPptEdit_GetPresentationId_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_GetPresentationId_InputMessage** request message, and the protocol server responds with an **IPptEdit_GetPresentationId_OutputMessage** response message.

3.1.4.27.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_GetPresentationId_InputMessage	The request WSDL message for the GetPresentationId WSDL operation.
IPptEdit_GetPresentationId_OutputMessage	The response WSDL message for the GetPresentationId WSDL operation.

3.1.4.27.1.1 IPptEdit_GetPresentationId_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/GetPresentationId
```

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

3.1.4.27.1.2 IPptEdit_GetPresentationId_OutputMessage

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

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

3.1.4.27.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetPresentationId	The input data for the GetPresentationId WSDL operation.
GetPresentationIdResponse	The result data for the GetPresentationId WSDL operation.

3.1.4.27.2.1 GetPresentationId

The **GetPresentationId** element specifies the input data for the **GetPresentationId** WSDL operation.

```
<xs:element name="GetPresentationId" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="rawDocId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="backendUriOverride" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

rawDocId: A **string** ([XMLSCHEMA2](#) section 3.2.1) that specifies the **absolute URL** to the presentation.

backendUriOverride: Reserved. MUST be ignored.

3.1.4.27.2.2 GetPresentationIdResponse

The **GetPresentationIdResponse** element specifies the result data for the **GetPresentationId** WSDL operation.

```
<xs:element name="GetPresentationIdResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
```

```

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

```

GetPresentationIdResult: A **string** ([XMLSCHEMA2](#) section 3.2.1) that specifies the identifier of the presentation.

3.1.4.27.3 Complex Types

None.

3.1.4.27.4 Simple Types

None.

3.1.4.27.5 Attributes

None.

3.1.4.27.6 Groups

None.

3.1.4.27.7 Attribute Groups

None.

3.1.4.28 InsertClipart

The **InsertClipart** operation SHOULD [<85>](#) be used to insert a picture into a presentation.

The following is the WSDL port type specification of the **InsertClipart** WSDL operation.

```

<wsdl:operation name="InsertClipart" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/InsertClipart"
message="tns4:IPptEdit_InsertClipart_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/InsertClipartResponse"
message="tns4:IPptEdit_InsertClipart_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptEdit_InsertClipart_InputMessage** request message, and the protocol server responds with an **IPptEdit_InsertClipart_OutputMessage** response message.

3.1.4.28.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_InsertClipart_InputMessage	The request WSDL message for the InsertClipart WSDL operation.
IPptEdit_InsertClipart_OutputMessage	The response WSDL message for the InsertClipart WSDL operation.

3.1.4.28.1.1 IPptEdit_InsertClipart_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/InsertClipart
```

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

3.1.4.28.1.2 IPptEdit_InsertClipart_OutputMessage

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

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

3.1.4.28.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
InsertClipart	The input data for the InsertClipart WSDL operation.
InsertClipartResponse	The result data for the InsertClipart WSDL operation.

3.1.4.28.2.1 InsertClipart

The **InsertClipart** element specifies the input data for the **InsertClipart** WSDL operation.

```
<xs:element name="InsertClipart" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="assetId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="Lcid" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="alt" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation into which the picture is inserted, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide into which the picture is inserted.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the placeholder shape into which the picture is inserted.

assetId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the unique identifier of the picture.

Lcid: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the **language code identifier (LCID)**, as specified in [\[MS-LCID\]](#), of the picture.

alt: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the alternate text for the picture.

3.1.4.28.2.2 InsertClipartResponse

The **InsertClipartResponse** element specifies the result data for the **InsertClipart** WSDL operation.

```
<xs:element name="InsertClipartResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="InsertClipartResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

InsertClipartResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.28.3 Complex Types

None.

3.1.4.28.4 Simple Types

None.

3.1.4.28.5 Attributes

None.

3.1.4.28.6 Groups

None.

3.1.4.28.7 Attribute Groups

None.

3.1.4.29 InsertShape

The **InsertShape** operation SHOULD [<86>](#) be used to insert a shape in a presentation.

The following is the WSDL port type specification of the **InsertShape** WSDL operation.

```
<wsdl:operation name="InsertShape" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
```

```

ServerInternalService/IPptEdit/InsertShape" message="tns4:IPptEdit_InsertShape_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/InsertShapeResponse"
message="tns4:IPptEdit_InsertShape_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptEdit_InsertShape_InputMessage** request message, and the protocol server responds with an **IPptEdit_InsertShape_OutputMessage** response message.

3.1.4.29.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_InsertShape_InputMessage	The request WSDL message for the InsertShape WSDL operation.
IPptEdit_InsertShape_OutputMessage	The response WSDL message for the InsertShape WSDL operation.

3.1.4.29.1.1 IPptEdit_InsertShape_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/InsertShape

```

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

3.1.4.29.1.2 IPptEdit_InsertShape_OutputMessage

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

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

3.1.4.29.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
InsertShape	The input data for the InsertShape WSDL operation.
InsertShapeResponse	The result data for the InsertShape WSDL operation.

3.1.4.29.2.1 InsertShape

The **InsertShape** element specifies the input data for the **InsertShape** WSDL operation.

```

<xs:element name="InsertShape" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeType" type="xs:int"/>
      <xs:element minOccurs="0" name="shapeCtrPosX" type="xs:int"/>
      <xs:element minOccurs="0" name="shapeCtrPosY" type="xs:int"/>
      <xs:element minOccurs="0" name="shapeWidth" type="xs:double"/>
      <xs:element minOccurs="0" name="shapeHeight" type="xs:double"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide.

shapeType: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the type of the shape that is inserted.

shapeCtrPosX: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the x coordinate of the shape that is inserted.

shapeCtrPosY: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the y coordinate of the shape that is inserted.

shapeWidth: A **double** ([\[XMLSCHEMA2\]](#) section 3.3.5) that specifies the width of the shape that is inserted.

shapeHeight: A **double** ([\[XMLSCHEMA2\]](#) section 3.3.5) that specifies the height of the shape that is inserted.

3.1.4.29.2 InsertShapeResponse

The **InsertShapeResponse** element specifies the result data for the **InsertShape** WSDL operation.

```

<xs:element name="InsertShapeResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="InsertShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

InsertShapeResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.29.3 Complex Types

None.

3.1.4.29.4 Simple Types

None.

3.1.4.29.5 Attributes

None.

3.1.4.29.6 Groups

None.

3.1.4.29.7 Attribute Groups

None.

3.1.4.30 InsertSlide

The **InsertSlide** operation is used to insert a presentation slide into a presentation.

The following is the WSDL port type specification of the **InsertSlide** WSDL operation.

```
<wsdl:operation name="InsertSlide" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/InsertSlide" message="tns4:IPptEdit_InsertSlide_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/InsertSlideResponse"
    message="tns4:IPptEdit_InsertSlide_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_InsertSlide_InputMessage** request message, and the protocol server responds with an **IPptEdit_InsertSlide_OutputMessage** response message.

3.1.4.30.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_InsertSlide_InputMessage	The request WSDL message for the InsertSlide WSDL operation.
IPptEdit_InsertSlide_OutputMessage	The response WSDL message for the InsertSlide WSDL operation.

3.1.4.30.1.1 IPptEdit_InsertSlide_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/InsertSlide
```

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

3.1.4.30.1.2 IPptEdit_InsertSlide_OutputMessage

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

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

3.1.4.30.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
InsertSlide	The input data for the InsertSlide WSDL operation.
InsertSlideResponse	The result data for the InsertSlide WSDL operation.

3.1.4.30.2.1 InsertSlide

The **InsertSlide** element specifies the input data for the **InsertSlide** WSDL operation.

```
<xs:element name="InsertSlide" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="dstId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="mainMasterId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="contentMasterId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

dstId: A **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide that precedes the location of the inserted slide. A value of zero indicates that no presentation slide precedes the location. In this case, the location of the inserted presentation slide will be the first slide of the presentation.

mainMasterId: An **ST_SlideMasterId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.16) that specifies the identifier for the main master slide of the inserted presentation slide.

contentMasterId: An **ST_SlideLayoutId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.14) that specifies the identifier for the slide layout of the inserted presentation slide.

3.1.4.30.2.2 InsertSlideResponse

The **InsertSlideResponse** element specifies the result data for the **InsertSlide** WSDL operation.

```
<xs:element name="InsertSlideResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="InsertSlideResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
```

</xs:element>

InsertSlideResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.30.3 Complex Types

None.

3.1.4.30.4 Simple Types

None.

3.1.4.30.5 Attributes

None.

3.1.4.30.6 Groups

None.

3.1.4.30.7 Attribute Groups

None.

3.1.4.31 InsertSmartArt

The **InsertSmartArt** operation is used to insert a SmartArt diagram into a presentation

The following is the WSDL port type specification of the **InsertSmartArt** WSDL operation.

```
<wsdl:operation name="InsertSmartArt" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/InsertSmartArt"
    message="tns4:IPptEdit_InsertSmartArt_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/InsertSmartArtResponse"
    message="tns4:IPptEdit_InsertSmartArt_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_InsertSmartArt_InputMessage** request message, and the protocol server responds with an **IPptEdit_InsertSmartArt_OutputMessage** response message.

3.1.4.31.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_InsertSmartArt_InputMessage	The request WSDL message for the InsertSmartArt WSDL operation.

Message	Description
IPptEdit_InsertSmartArt_OutputMessage	The response WSDL message for the InsertSmartArt WSDL operation.

3.1.4.31.1.1 IPptEdit_InsertSmartArt_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/InsertSmartArt
```

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

3.1.4.31.1.2 IPptEdit_InsertSmartArt_OutputMessage

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

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

3.1.4.31.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
InsertSmartArt	The input data for the InsertSmartArt WSDL operation.
InsertSmartArtResponse	The result data for the InsertSmartArt WSDL operation.

3.1.4.31.2.1 InsertSmartArt

The **InsertSmartArt** element specifies the input data for the **InsertSmartArt** WSDL operation.

```
<xs:element name="InsertSmartArt" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="layoutUri" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="category" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation into which the SmartArt diagram is inserted, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide into which the SmartArt diagram is inserted.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the placeholder shape into which the SmartArt diagram is inserted.

layoutUri: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the layout with which the SmartArt diagram is inserted.

category: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the category to which the layout belongs.

3.1.4.31.2 InsertSmartArtResponse

The **InsertSmartArtResponse** element specifies the result data for the **InsertSmartArt** WSDL operation.

```
<xs:element name="InsertSmartArtResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="InsertSmartArtResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

InsertSmartArtResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.31.3 Complex Types

None.

3.1.4.31.4 Simple Types

None.

3.1.4.31.5 Attributes

None.

3.1.4.31.6 Groups

None.

3.1.4.31.7 Attribute Groups

None.

3.1.4.32 LogULS

Reserved. MUST not be called.

The following is the WSDL port type specification of the **LogULS** WSDL operation.

```
<wsdl:operation name="LogULS" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/LogULS" message="tns4:IPptEdit_LogULS_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
```



```
ServerInternalService/IPptEdit/LogULSResponse" message="tns4:IPptEdit_LogULS_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

Reserved. MUST not be called.

3.1.4.32.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_LogULS_InputMessage	The request WSDL message for the LogULS WSDL operation.
IPptEdit_LogULS_OutputMessage	The response WSDL message for the LogULS WSDL operation.

3.1.4.32.1.1 IPptEdit_LogULS_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/LogULS
```

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

3.1.4.32.1.2 IPptEdit_LogULS_OutputMessage

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

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

3.1.4.32.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
LogULS	The input data for the LogULS WSDL operation.
LogULSResponse	The result data for the LogULS WSDL operation.

3.1.4.32.2.1 LogULS

The **LogULS** element specifies the input data for the **LogULS** WSDL operation.

```
<xs:element name="LogULS" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="tagID" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="level" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

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

tagID: Reserved. MUST be ignored.

level: Reserved. MUST be ignored.

message: Reserved. MUST be ignored.

3.1.4.32.2 LogULSResponse

The **LogULSResponse** element specifies the result data for the **LogULS** WSDL operation.

```
<xs:element name="LogULSResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="LogULSResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

LogULSResult: Reserved. MUST be ignored.

3.1.4.32.3 Complex Types

None.

3.1.4.32.4 Simple Types

None.

3.1.4.32.5 Attributes

None.

3.1.4.32.6 Groups

None.

3.1.4.32.7 Attribute Groups

None.

3.1.4.33 MoveComment

The **MoveComment** operation SHOULD [<87>](#) be used to change the position of a comment in a presentation.

The following is the WSDL port type specification of the **MoveComment** WSDL operation.

```
<wsdl:operation name="MoveComment" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
```

```

ServerInternalService/IPptEdit/MoveComment" message="tns4:IPptEdit_MoveComment_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/MoveCommentResponse"
message="tns4:IPptEdit_MoveComment_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptEdit_MoveComment_InputMessage** request message, and the protocol server responds with an **IPptEdit_MoveComment_OutputMessage** response message.

3.1.4.33.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_MoveComment_InputMessage	The request WSDL message for the MoveComment WSDL operation.
IPptEdit_MoveComment_OutputMessage	The response WSDL message for the MoveComment WSDL operation.

3.1.4.33.1.1 IPptEdit_MoveComment_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/MoveComment

```

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

3.1.4.33.1.2 IPptEdit_MoveComment_OutputMessage

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

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

3.1.4.33.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
MoveComment	The input data for the MoveComment WSDL operation.
MoveCommentResponse	The result data for the MoveComment WSDL operation.

3.1.4.33.2.1 MoveComment

The **MoveComment** element specifies the input data for the **MoveComment** WSDL operation.

```
<xs:element name="MoveComment" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="authorId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="index" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="x" type="xs:double"/>
      <xs:element minOccurs="0" name="y" type="xs:double"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the comment that is being modified.

authorId: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the identifier of the author of the comment.

index: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the identifier of the comment within its author's comments.

x: A **double** ([\[XMLSCHEMA2\]](#) section 3.3.5) that specifies the new x coordinate of the comment.

y: A **double** ([\[XMLSCHEMA2\]](#) section 3.3.5) that specifies the new y coordinate of the comment.

3.1.4.33.2 MoveCommentResponse

The **MoveCommentResponse** element specifies the result data for the **MoveComment** WSDL operation.

```
<xs:element name="MoveCommentResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="MoveCommentResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

MoveCommentResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.33.3 Complex Types

None.

3.1.4.33.4 Simple Types

None.

3.1.4.33.5 Attributes

None.

3.1.4.33.6 Groups

None.

3.1.4.33.7 Attribute Groups

None.

3.1.4.34 MoveShape

The **MoveShape** operation SHOULD [<88>](#) be used to move a shape in a presentation.

The following is the WSDL port type specification of the **MoveShape** WSDL operation.

```
<wsdl:operation name="MoveShape" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/MoveShape" message="tns4:IPptEdit_MoveShape_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/MoveShapeResponse"
    message="tns4:IPptEdit_MoveShape_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_MoveShape_InputMessage** request message, and the protocol server responds with an **IPptEdit_MoveShape_OutputMessage** response message.

3.1.4.34.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_MoveShape_InputMessage	The request WSDL message for the MoveShape WSDL operation.
IPptEdit_MoveShape_OutputMessage	The response WSDL message for the MoveShape WSDL operation.

3.1.4.34.1.1 IPptEdit_MoveShape_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/MoveShape
```

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

3.1.4.34.1.2 IPptEdit_MoveShape_OutputMessage

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

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

3.1.4.34.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
MoveShape	The input data for the MoveShape WSDL operation.
MoveShapeResponse	The result data for the MoveShape WSDL operation.

3.1.4.34.2.1 MoveShape

The **MoveShape** element specifies the input data for the **MoveShape** WSDL operation.

```
<xs:element name="MoveShape" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="x" type="xs:double"/>
      <xs:element minOccurs="0" name="y" type="xs:double"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape that is being modified.

x: A **double** ([XMLSCHEMA2] section 3.3.5) that specifies the new x coordinate of the shape.

y: A **double** ([XMLSCHEMA2] section 3.3.5) that specifies the new y coordinate of the shape.

3.1.4.34.2.2 MoveShapeResponse

The **MoveShapeResponse** element specifies the result data for the **MoveShape** WSDL operation.

```
<xs:element name="MoveShapeResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
        name="MoveShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

MoveShapeResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.34.3 Complex Types

None.

3.1.4.34.4 Simple Types

None.

3.1.4.34.5 Attributes

None.

3.1.4.34.6 Groups

None.

3.1.4.34.7 Attribute Groups

None.

3.1.4.35 MoveSlide

The **MoveSlide** operation is used to move a presentation slide in a presentation.

The following is the WSDL port type specification of the **MoveSlide** WSDL operation.

```
<wsdl:operation name="MoveSlide" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/MoveSlide" message="tns4:IPptEdit MoveSlide InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/MoveSlideResponse"
    message="tns4:IPptEdit_MoveSlide_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_MoveSlide_InputMessage** request message, and the protocol server responds with an **IPptEdit_MoveSlide_OutputMessage** response message.

3.1.4.35.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_MoveSlide_InputMessage	The request WSDL message for the MoveSlide WSDL operation.
IPptEdit_MoveSlide_OutputMessage	The response WSDL message for the MoveSlide WSDL operation.

3.1.4.35.1.1 IPptEdit_MoveSlide_InputMessage

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

The SOAP action value is:

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

3.1.4.35.1.2 IPptEdit_MoveSlide_OutputMessage

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

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

3.1.4.35.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
MoveSlide	The input data for the MoveSlide WSDL operation.
MoveSlideResponse	The result data for the MoveSlide WSDL operation.

3.1.4.35.2.1 MoveSlide

The **MoveSlide** element specifies the input data for the **MoveSlide** WSDL operation.

```
<xs:element name="MoveSlide" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="srcId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="dstId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the slide that is moved, as specified in section [3.1.4.27](#).

srcId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide that is moved.

dstId: A value of zero or an **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide that precedes the new location of the moved slide. A value of zero indicates that no presentation slide precedes the new location. In this case, the new location of the moved presentation slide is the first slide of the presentation.

3.1.4.35.2.2 MoveSlideResponse

The **MoveSlideResponse** element specifies the result data for the **MoveSlide** WSDL operation.

```
<xs:element name="MoveSlideResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
        name="MoveSlideResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```



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

MoveSlideResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.35.3 Complex Types

None.

3.1.4.35.4 Simple Types

None.

3.1.4.35.5 Attributes

None.

3.1.4.35.6 Groups

None.

3.1.4.35.7 Attribute Groups

None.

3.1.4.36 PastePicture

The **PastePicture** operation is used to paste a picture into a presentation.

The following is the WSDL port type specification of the **PastePicture** WSDL operation.

```
<wsdl:operation name="PastePicture" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/PastePicture"
    message="tns4:IPptEdit_PastePicture_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/PastePictureResponse"
    message="tns4:IPptEdit_PastePicture_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_PastePicture_InputMessage** request message, and the protocol server responds with an **IPptEdit_PastePicture_OutputMessage** response message

3.1.4.36.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_PastePicture_InputMessage	The request WSDL message for the PastePicture WSDL operation.

Message	Description
IPptEdit_PastePicture_OutputMessage	The response WSDL message for the PastePicture WSDL operation.

3.1.4.36.1.1 IPptEdit_PastePicture_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/PastePicture
```

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

3.1.4.36.1.2 IPptEdit_PastePicture_OutputMessage

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

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

3.1.4.36.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
PastePicture	The input data for the PastePicture WSDL operation.
PastePictureResponse	The result data for the PastePicture WSDL operation.

3.1.4.36.2.1 PastePicture

The **PastePicture** element specifies the input data for the **PastePicture** WSDL operation.

```
<xs:element name="PastePicture" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="url" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="alt" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation into which the picture is inserted, as specified in section [3.1.4.27](#).

slideId: **slideId:** An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide into which the picture is inserted.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the placeholder shape into which the picture is inserted.

url: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the location of the picture.

alt: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the alternate text for the picture.

3.1.4.36.2 PastePictureResponse

The **PastePictureResponse** element specifies the result data for the **PastePicture** WSDL operation.

```
<xs:element name="PastePictureResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="PastePictureResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

PastePictureResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.36.3 Complex Types

None.

3.1.4.36.4 Simple Types

None.

3.1.4.36.5 Attributes

None.

3.1.4.36.6 Groups

None.

3.1.4.36.7 Attribute Groups

None.

3.1.4.37 Print

A **Print** operation **SHOULD** [<89>](#) be used to print a presentation.

The following is the WSDL port type specification of the **Print** WSDL operation.

```
<wsdl:operation name="Print" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input>
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/Print" message="tns4:IPptEdit_Print_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output>
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/PrintResponse" message="tns4:IPptEdit_Print_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

</wsdl:operation>

The protocol client sends an **IPptEdit_Print_InputMessage** request message, and the protocol server responds with an **IPptEdit_Print_OutputMessage** response message.

3.1.4.37.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_Print_InputMessage	The request WSDL message for the Print WSDL operation.
IPptEdit_Print_OutputMessage	The response WSDL message for the Print WSDL operation.

3.1.4.37.1.1 IPptEdit_Print_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/Print
```

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

3.1.4.37.1.2 IPptEdit_Print_OutputMessage

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

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

3.1.4.37.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
Print	The input data for the Print WSDL operation.
PrintResponse	The result data for the Print WSDL operation.

3.1.4.37.2.1 Print

The **Print** element specifies the input data for the **Print** WSDL operation.

```
<xs:element name="Print" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>  
    </xs:sequence>  
  </xs:element>
```

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

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation to print, as specified in section [3.1.4.27](#).

3.1.4.37.2 PrintResponse

The **PrintResponse** element specifies the result data for the **Print** WSDL operation.

```
<xs:element name="PrintResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="PrintResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

PrintResult: A **PptViewingService.PrintResult** (section [2.2.4.39](#)) that specifies the result of the operation.

3.1.4.37.3 Complex Types

None.

3.1.4.37.4 Simple Types

None.

3.1.4.37.5 Attributes

None.

3.1.4.37.6 Groups

None.

3.1.4.37.7 Attribute Groups

None.

3.1.4.38 Redo

The **Redo** operation is used to redo the last operation in a presentation that is undone.

The following is the WSDL port type specification of the **Redo** WSDL operation.

```
<wsdl:operation name="Redo" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/Redo" message="tns4:IPptEdit_Red_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/RedoResponse" message="tns4:IPptEdit_Red_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
```

</wsdl:operation>

The protocol client sends an **IPptEdit_Redo_InputMessage** request message, and the protocol server responds with an **IPptEdit_Redo_OutputMessage** response message.

3.1.4.38.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_Redo_InputMessage	The request WSDL message for the Redo WSDL operation.
IPptEdit_Redo_OutputMessage	The response WSDL message for the Redo WSDL operation.

3.1.4.38.1.1 IPptEdit_Redo_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/Redo
```

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

3.1.4.38.1.2 IPptEdit_Redo_OutputMessage

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

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

3.1.4.38.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
Redo	The input data for the Redo WSDL operation.
RedoResponse	The result data for the Redo WSDL operation.

3.1.4.38.2.1 Redo

The **Redo** element specifies the input data for the **Redo** WSDL operation.

```
<xs:element name="Redo" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

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

presentationId: A string ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

3.1.4.38.2 RedoResponse

The **RedoResponse** element specifies the result data for the **Redo** WSDL operation.

```
<xs:element name="RedoResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="RedoResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

RedoResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.38.3 Complex Types

None.

3.1.4.38.4 Simple Types

None.

3.1.4.38.5 Attributes

None.

3.1.4.38.6 Groups

None.

3.1.4.38.7 Attribute Groups

None.

3.1.4.39 RemoveShapeFill

The **RemoveShapeFill** operation SHOULD [<90>](#) be used to remove the color of a shape in a presentation.

The following is the WSDL port type specification of the **RemoveShapeFill** WSDL operation.

```
<wsdl:operation name="RemoveShapeFill" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/RemoveShapeFill"
message="tns4:IPptEdit_RemoveShapeFill_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
```

```

ServerInternalService/IPptEdit/RemoveShapeFillResponse"
message="tns4:IPptEdit RemoveShapeFill OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptEdit_RemoveShapeFill_InputMessage** request message, and the protocol server responds with an **IPptEdit_RemoveShapeFill_OutputMessage** response message.

3.1.4.39.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_RemoveShapeFill_InputMessage	The request WSDL message for the RemoveShapeFill WSDL operation.
IPptEdit_RemoveShapeFill_OutputMessage	The response WSDL message for the RemoveShapeFill WSDL operation.

3.1.4.39.1.1 IPptEdit_RemoveShapeFill_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/RemoveShapeFill

```

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

3.1.4.39.1.2 IPptEdit_RemoveShapeFill_OutputMessage

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

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

3.1.4.39.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
RemoveShapeFill	The input data for the RemoveShapeFill WSDL operation.
RemoveShapeFillResponse	The result data for the RemoveShapeFill WSDL operation.

3.1.4.39.2.1 RemoveShapeFill

The **RemoveShapeFill** element specifies the input data for the **RemoveShapeFill** WSDL operation.

```

<xs:element name="RemoveShapeFill" xmlns:xs="http://www.w3.org/2001/XMLSchema">

```



```

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

```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape that is being modified.

3.1.4.39.2 RemoveShapeFillResponse

The **RemoveShapeFillResponse** element specifies the result data for the **RemoveShapeFill** WSDL operation.

```

<xs:element name="RemoveShapeFillResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="RemoveShapeFillResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

RemoveShapeFillResult: A **ServiceResult** (section [2.2.4.43](#)) element that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)) element.

3.1.4.39.3 Complex Types

None.

3.1.4.39.4 Simple Types

None.

3.1.4.39.5 Attributes

None.

3.1.4.39.6 Groups

None.

3.1.4.39.7 Attribute Groups

None.

3.1.4.40 RemoveShapeOutline

The **RemoveShapeOutline** operation SHOULD [<91>](#) be used to remove the outline of a shape in a presentation.

The following is the WSDL port type specification of the **RemoveShapeOutline** WSDL operation.

```
<wsdl:operation name="RemoveShapeOutline" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/RemoveShapeOutline"
    message="tns4:IPptEdit_RemoveShapeOutline_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/RemoveShapeOutlineResponse"
    message="tns4:IPptEdit_RemoveShapeOutline_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_RemoveShapeOutline_InputMessage** request message, and the protocol server responds with an **IPptEdit_RemoveShapeOutline_OutputMessage** response message.

3.1.4.40.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_RemoveShapeOutline_InputMessage	The request WSDL message for the RemoveShapeOutline WSDL operation.
IPptEdit_RemoveShapeOutline_OutputMessage	The response WSDL message for the RemoveShapeOutline WSDL operation.

3.1.4.40.1.1 IPptEdit_RemoveShapeOutline_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/RemoveShapeOutline
```

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

3.1.4.40.1.2 IPptEdit_RemoveShapeOutline_OutputMessage

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

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

3.1.4.40.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
RemoveShapeOutline	The input data for the RemoveShapeOutline WSDL operation.
RemoveShapeOutlineResponse	The result data for the RemoveShapeOutline WSDL operation.

3.1.4.40.2.1 RemoveShapeOutline

The **RemoveShapeOutline** element specifies the input data for the **RemoveShapeOutline** WSDL operation.

```
<xs:element name="RemoveShapeOutline" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the shape that is being modified.

3.1.4.40.2.2 RemoveShapeOutlineResponse

The **RemoveShapeOutlineResponse** element specifies the result data for the **RemoveShapeOutline** WSDL operation.

```
<xs:element name="RemoveShapeOutlineResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="RemoveShapeOutlineResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

RemoveShapeOutlineResult: A **ServiceResult** (section [2.2.4.43](#)) element that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.40.3 Complex Types

None.

3.1.4.40.4 Simple Types

None.

3.1.4.40.5 Attributes

None.

3.1.4.40.6 Groups

None.

3.1.4.40.7 Attribute Groups

None.

3.1.4.41 ReorderAnimation

The **ReorderAnimation** operation SHOULD [<92>](#) be used to reorder the animation of a shape in a presentation.

The following is the WSDL port type specification of the **ReorderAnimation** WSDL operation.

```
<wsdl:operation name="ReorderAnimation" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ReorderAnimation"
    message="tns4:IPptEdit_ReorderAnimation_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ReorderAnimationResponse"
    message="tns4:IPptEdit_ReorderAnimation_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ReorderAnimation_InputMessage** request message, and the protocol server responds with an **IPptEdit_ReorderAnimation_OutputMessage** response message.

3.1.4.41.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ReorderAnimation_InputMessage	The request WSDL message for the ReorderAnimation WSDL operation.
IPptEdit_ReorderAnimation_OutputMessage	The response WSDL message for the ReorderAnimation WSDL operation.

3.1.4.41.1.1 IPptEdit_ReorderAnimation_InputMessage

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

The SOAP action value is:

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

3.1.4.41.1.2 IPptEdit_ReorderAnimation_OutputMessage

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

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

3.1.4.41.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ReorderAnimation	The input data for the ReorderAnimation WSDL operation.
ReorderAnimationResponse	The result data for the ReorderAnimation WSDL operation.

3.1.4.41.2.1 ReorderAnimation

The **ReorderAnimation** element specifies the input data for the **ReorderAnimation** WSDL operation.

```
<xs:element name="ReorderAnimation" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="fLater" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the shape. The shape **MUST** have exactly one effect and that effect **MUST** exist on the main timeline.

fLater: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that specifies whether to move the effect earlier or later in the timeline. If this value is true, the effect **MUST NOT** have the largest build identifier. If this value is false, the effect **MUST NOT** have the smallest build identifier.

3.1.4.41.2.2 ReorderAnimationResponse

The **ReorderAnimationResponse** element specifies the result data for the **ReorderAnimation** WSDL operation.

```

<xs:element name="ReorderAnimationResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ReorderAnimationResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

ReorderAnimationResult: A **ServiceResult** (section [2.2.4.43](#)) element that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.41.3 Complex Types

None.

3.1.4.41.4 Simple Types

None.

3.1.4.41.5 Attributes

None.

3.1.4.41.6 Groups

None.

3.1.4.41.7 Attribute Groups

None.

3.1.4.42 ReplaceNotes

The **ReplaceNotes** operation is used to replace presentation notes in a presentation.

The following is the WSDL port type specification of the **ReplaceNotes** WSDL operation.

```

<wsdl:operation name="ReplaceNotes" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReplaceNotes"
message="tns4:IPptEdit ReplaceNotes InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReplaceNotesResponse"
message="tns4:IPptEdit ReplaceNotes OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptEdit_ReplaceNotes_InputMessage** request message, and the protocol server responds with an **IPptEdit_ReplaceNotes_OutputMessage** response message.

3.1.4.42.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ReplaceNotes_InputMessage	The request WSDL message for the ReplaceNotes WSDL operation.
IPptEdit_ReplaceNotes_OutputMessage	The response WSDL message for the ReplaceNotes WSDL operation.

3.1.4.42.1.1 IPptEdit_ReplaceNotes_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ReplaceNotes
```

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

3.1.4.42.1.2 IPptEdit_ReplaceNotes_OutputMessage

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

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

3.1.4.42.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ReplaceNotes	The input data for the ReplaceNotes WSDL operation.
ReplaceNotesResponse	The result data for the ReplaceNotes WSDL operation.

3.1.4.42.2.1 ReplaceNotes

The **ReplaceNotes** element specifies the input data for the **ReplaceNotes** WSDL operation.

```
<xs:element name="ReplaceNotes" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="newNotes" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the text that is replaced, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the text that is replaced.

newNotes: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the replacement notes.

3.1.4.42.2 ReplaceNotesResponse

The **ReplaceNotesResponse** element specifies the result data for the **ReplaceNotes** WSDL operation.

```
<xs:element name="ReplaceNotesResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ReplaceNotesResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ReplaceNotesResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.42.3 Complex Types

None.

3.1.4.42.4 Simple Types

None.

3.1.4.42.5 Attributes

None.

3.1.4.42.6 Groups

None.

3.1.4.42.7 Attribute Groups

None.

3.1.4.43 ReplaceText

The **ReplaceText** operation is used to replace text in a shape in a presentation.

The following is the WSDL port type specification of the **ReplaceText** WSDL operation.

```
<wsdl:operation name="ReplaceText" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReplaceText" message="tns4:IPptEdit_ReplaceText_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReplaceTextResponse"
message="tns4:IPptEdit_ReplaceText_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```


The protocol client sends an **IPptEdit_ReplaceText_InputMessage** request message, and the protocol server responds with an **IPptEdit_ReplaceText_OutputMessage** response message.

3.1.4.43.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ReplaceText_InputMessage	The request WSDL message for the ReplaceText WSDL operation.
IPptEdit_ReplaceText_OutputMessage	The response WSDL message for the ReplaceText WSDL operation.

3.1.4.43.1.1 IPptEdit_ReplaceText_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ReplaceText
```

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

3.1.4.43.1.2 IPptEdit_ReplaceText_OutputMessage

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

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

3.1.4.43.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ReplaceText	The input data for the ReplaceText WSDL operation.
ReplaceTextResponse	The result data for the ReplaceText WSDL operation.

3.1.4.43.2.1 ReplaceText

The **ReplaceText** element specifies the input data for the **ReplaceText** WSDL operation.

```
<xs:element name="ReplaceText" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="newText" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="partialEdit" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

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

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the text that is replaced, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the text that is replaced.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the shape containing the text that is replaced.

newText: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the replacement text.

partialEdit: Reserved. MUST be ignored.

3.1.4.43.2 ReplaceTextResponse

The **ReplaceTextResponse** element specifies the result data for the **ReplaceText** WSDL operation.

```
<xs:element name="ReplaceTextResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ReplaceTextResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ReplaceTextResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.43.3 Complex Types

None.

3.1.4.43.4 Simple Types

None.

3.1.4.43.5 Attributes

None.

3.1.4.43.6 Groups

None.

3.1.4.43.7 Attribute Groups

None.

3.1.4.44 ResetPicture

The **ResetPicture** operation SHOULD [<93>](#) be used to remove all customizations on a picture in a presentation.

The following is the WSDL port type specification of the **ResetPicture** WSDL operation.

```
<wsdl:operation name="ResetPicture" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ResetPicture"
    message="tns4:IPptEdit_ResetPicture_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ResetPictureResponse"
    message="tns4:IPptEdit_ResetPicture_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ResetPicture_InputMessage** request message, and the protocol server responds with an **IPptEdit_ResetPicture_OutputMessage** response message.

3.1.4.44.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ResetPicture_InputMessage	The request WSDL message for the ResetPicture WSDL operation.
IPptEdit_ResetPicture_OutputMessage	The response WSDL message for the ResetPicture WSDL operation.

3.1.4.44.1.1 IPptEdit_ResetPicture_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ResetPicture
```

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

3.1.4.44.1.2 IPptEdit_ResetPicture_OutputMessage

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

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

3.1.4.44.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ResetPicture	The input data for the ResetPicture WSDL operation.
ResetPictureResponse	The result data for the ResetPicture WSDL operation.

3.1.4.44.2.1 ResetPicture

The **ResetPicture** element specifies the input data for the **ResetPicture** WSDL operation.

```
<xs:element name="ResetPicture" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

slideId: A **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the picture that is being modified.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the picture that is being modified.

3.1.4.44.2.2 ResetPictureResponse

The **ResetPictureResponse** element specifies the result data for the **ResetPicture** WSDL operation.

```
<xs:element name="ResetPictureResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ResetPictureResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ResetPictureResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.44.3 Complex Types

None.

3.1.4.44.4 Simple Types

None.

3.1.4.44.5 Attributes

None.

3.1.4.44.6 Groups

None.

3.1.4.44.7 Attribute Groups

None.

3.1.4.45 ResetSmartArt

The **ResetSmartArt** operation is used to remove all customizations on a SmartArt diagram in a presentation.

The following is the WSDL port type specification of the **ResetSmartArt** WSDL operation.

```
<wsdl:operation name="ResetSmartArt" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ResetSmartArt"
    message="tns4:IPptEdit_ResetSmartArt_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ResetSmartArtResponse"
    message="tns4:IPptEdit_ResetSmartArt_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ResetSmartArt_InputMessage** request message, and the protocol server responds with an **IPptEdit_ResetSmartArt_OutputMessage** response message.

3.1.4.45.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ResetSmartArt_InputMessage	The request WSDL message for the ResetSmartArt WSDL operation.
IPptEdit_ResetSmartArt_OutputMessage	The response WSDL message for the ResetSmartArt WSDL operation.

3.1.4.45.1.1 IPptEdit_ResetSmartArt_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ResetSmartArt
```

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

3.1.4.45.1.2 IPptEdit_ResetSmartArt_OutputMessage

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

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

3.1.4.45.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ResetSmartArt	The input data for the ResetSmartArt WSDL operation.
ResetSmartArtResponse	The result data for the ResetSmartArt WSDL operation.

3.1.4.45.2.1 ResetSmartArt

The **ResetSmartArt** element specifies the input data for the **ResetSmartArt** WSDL operation.

```
<xs:element name="ResetSmartArt" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the SmartArt diagram that is reset, as specified in section [3.1.4.27](#).

slideId: A **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the SmartArt diagram that is reset.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the SmartArt diagram that is reset.

3.1.4.45.2.2 ResetSmartArtResponse

The **ResetSmartArtResponse** element specifies the result data for the **ResetSmartArt** WSDL operation.

```
<xs:element name="ResetSmartArtResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ResetSmartArtResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ResetSmartArtResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.45.3 Complex Types

None.

3.1.4.45.4 Simple Types

None.

3.1.4.45.5 Attributes

None.

3.1.4.45.6 Groups

None.

3.1.4.45.7 Attribute Groups

None.

3.1.4.46 ResizeShape

The **ResizeShape** operation SHOULD [<94>](#) be used to resize a shape in a presentation.

The following is the WSDL port type specification of the **ResizeShape** WSDL operation.

```
<wsdl:operation name="ResizeShape" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ResizeShape" message="tns4:IPptEdit_ResizeShape_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ResizeShapeResponse"
    message="tns4:IPptEdit_ResizeShape_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ResizeShape_InputMessage** request message, and the protocol server responds with an **IPptEdit_ResizeShape_OutputMessage** response message.

3.1.4.46.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ResizeShape_InputMessage	The request WSDL message for the ResizeShape WSDL operation.
IPptEdit_ResizeShape_OutputMessage	The response WSDL message for the ResizeShape WSDL operation.

3.1.4.46.1.1 IPptEdit_ResizeShape_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ResizeShape
```

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

3.1.4.46.1.2 IPptEdit_ResizeShape_OutputMessage

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

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

3.1.4.46.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ResizeShape	The input data for the ResizeShape WSDL operation.
ResizeShapeResponse	The result data for the ResizeShape WSDL operation.

3.1.4.46.2.1 ResizeShape

The **ResizeShape** element specifies the input data for the **ResizeShape** WSDL operation.

```
<xs:element name="ResizeShape" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="deltaNorth" type="xs:double"/>
      <xs:element minOccurs="0" name="deltaSouth" type="xs:double"/>
      <xs:element minOccurs="0" name="deltaWest" type="xs:double"/>
      <xs:element minOccurs="0" name="deltaEast" type="xs:double"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([ISO/IEC29500-1:2011] section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape that is being modified.

deltaNorth: A **double** ([XMLSCHEMA2] section 3.3.5) that specifies how the shape is resized. If the shape is a line, then the start of the y coordinate of the line is offset by this value. Else the north edge of the shape is offset by this value.

deltaSouth: A **double** ([XMLSCHEMA2] section 3.3.5) that specifies how the shape is resized. If the shape is a line, then the start of the x coordinate of the line is offset by this value. Else the south edge of the shape is offset by this value.

deltaWest: A **double** ([XMLSCHEMA2] section 3.3.5) that specifies how the shape is resized. If the shape is a line, then the end of the y coordinate of the line is offset by this value. Else the west edge of the shape is offset by this value.

deltaEast: A **double** ([XMLSCHEMA2] section 3.3.5) that specifies how the shape is resized. If the shape is a line, then the end of the x coordinate of the line is offset by this value. Else the east edge of the shape is offset by this value.

3.1.4.46.2 ResizeShapeResponse

The **ResizeShapeResponse** element specifies the result data for the **ResizeShape** WSDL operation.

```
<xs:element name="ResizeShapeResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ResizeShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ResizeShapeResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.46.3 Complex Types

None.

3.1.4.46.4 Simple Types

None.

3.1.4.46.5 Attributes

None.

3.1.4.46.6 Groups

None.

3.1.4.46.7 Attribute Groups

None.

3.1.4.47 ReverseSmartArt

The **ReverseSmartArt** operation is used to reverse the direction of a SmartArt diagram in a presentation.

The following is the WSDL port type specification of the **ReverseSmartArt** WSDL operation.

```
<wsdl:operation name="ReverseSmartArt" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReverseSmartArt"
message="tns4:IPptEdit ReverseSmartArt InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReverseSmartArtResponse"
message="tns4:IPptEdit_ReverseSmartArt_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ReverseSmartArt_InputMessage** request message, and the protocol server responds with an **IPptEdit_ReverseSmartArt_OutputMessage** response message.

3.1.4.47.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ReverseSmartArt_InputMessage	The request WSDL message for the ReverseSmartArt WSDL operation.
IPptEdit_ReverseSmartArt_OutputMessage	The response WSDL message for the ReverseSmartArt WSDL operation.

3.1.4.47.1.1 IPptEdit_ReverseSmartArt_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ReverseSmartArt
```

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

3.1.4.47.1.2 IPptEdit_ReverseSmartArt_OutputMessage

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

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

3.1.4.47.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ReverseSmartArt	The input data for the ReverseSmartArt WSDL operation.
ReverseSmartArtResponse	The result data for the ReverseSmartArt WSDL operation.

3.1.4.47.2.1 ReverseSmartArt

The **ReverseSmartArt** element specifies the input data for the **ReverseSmartArt** WSDL operation.

```
<xs:element name="ReverseSmartArt" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>  
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>  
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the SmartArt diagram that is reversed, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the SmartArt diagram that is reversed.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the SmartArt diagram that is reversed.

3.1.4.47.2 ReverseSmartArtResponse

The **ReverseSmartArtResponse** element specifies the result data for the **ReverseSmartArt** WSDL operation.

```
<xs:element name="ReverseSmartArtResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ReverseSmartArtResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ReverseSmartArtResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.47.3 Complex Types

None.

3.1.4.47.4 Simple Types

None.

3.1.4.47.5 Attributes

None.

3.1.4.47.6 Groups

None.

3.1.4.47.7 Attribute Groups

None.

3.1.4.48 RotateShape

The **RotateShape** operation SHOULD [<95>](#) be used to rotate a shape in a presentation.

The following is the WSDL port type specification of the **RotateShape** WSDL operation.

```
<wsdl:operation name="RotateShape" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/RotateShape" message="tns4:IPptEdit_RotateShape_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
```

```

ServerInternalService/IPptEdit/RotateShapeResponse"
message="tns4:IPptEdit_RotateShape_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptEdit_RotateShape_InputMessage** request message, and the protocol server responds with an **IPptEdit_RotateShape_OutputMessage** response message.

3.1.4.48.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_RotateShape_InputMessage	The request WSDL message for the RotateShape WSDL operation.
IPptEdit_RotateShape_OutputMessage	The response WSDL message for the RotateShape WSDL operation.

3.1.4.48.1.1 IPptEdit_RotateShape_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/RotateShape

```

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

3.1.4.48.1.2 IPptEdit_RotateShape_OutputMessage

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

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

3.1.4.48.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
RotateShape	The input data for the RotateShape WSDL operation.
RotateShapeResponse	The result data for the RotateShape WSDL operation.

3.1.4.48.2.1 RotateShape

The **RotateShape** element specifies the input data for the **RotateShape** WSDL operation.

```

<xs:element name="RotateShape" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>

```

```

<xs:sequence>
  <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
  <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
  <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
  <xs:element minOccurs="0" name="degrees" type="xs:int"/>
</xs:sequence>
</xs:complexType>
</xs:element>

```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the shape that is being modified. The shape **MUST NOT** be a chart, SmartArt, table, or ActiveX shape.

degrees: An **int** ([\[XMLSCHEMA2\]](#) section 3.3.17) that specifies the number of degrees the shape is rotated from its default orientation.

3.1.4.48.2 RotateShapeResponse

The **RotateShapeResponse** element specifies the result data for the **RotateShape** WSDL operation.

```

<xs:element name="RotateShapeResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="RotateShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

RotateShapeResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.48.3 Complex Types

None.

3.1.4.48.4 Simple Types

None.

3.1.4.48.5 Attributes

None.

3.1.4.48.6 Groups

None.

3.1.4.48.7 Attribute Groups

None.

3.1.4.49 SaveAndClose

The **SaveAndClose** operation is used to save the contents of a presentation and unlock it.

The following is the WSDL port type specification of the **SaveAndClose** WSDL operation.

```
<wsdl:operation name="SaveAndClose" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SaveAndClose"
    message="tns4:IPptEdit_SaveAndClose_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SaveAndCloseResponse"
    message="tns4:IPptEdit_SaveAndClose_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_SaveAndClose_InputMessage** request message, and the protocol server responds with an **IPptEdit_SaveAndClose_OutputMessage** response message.

3.1.4.49.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SaveAndClose_InputMessage	The request WSDL message for the SaveAndClose WSDL operation.
IPptEdit_SaveAndClose_OutputMessage	The response WSDL message for the SaveAndClose WSDL operation.

3.1.4.49.1.1 IPptEdit_SaveAndClose_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SaveAndClose
```

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

3.1.4.49.1.2 IPptEdit_SaveAndClose_OutputMessage

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

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

3.1.4.49.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SaveAndClose	The input data for the SaveAndClose WSDL operation.
SaveAndCloseResponse	The result data for the SaveAndClose WSDL operation.

3.1.4.49.2.1 SaveAndClose

The **SaveAndClose** element specifies the input data for the **SaveAndClose** WSDL operation.

```
<xs:element name="SaveAndClose" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="fReleaseLock" type="xs:boolean"/>
      <xs:element minOccurs="0" name="fWaitForSave" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

fReleaseLock: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that specifies whether the presentation will become unlocked.

fWaitForSave: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that specifies whether the protocol server makes a blocking call until the save is finished.

3.1.4.49.2.2 SaveAndCloseResponse

The **SaveAndCloseResponse** element specifies the result data for the **SaveAndClose** WSDL operation.

```
<xs:element name="SaveAndCloseResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SaveAndCloseResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SaveAndCloseResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.49.3 Complex Types

None.

3.1.4.49.4 Simple Types

None.

3.1.4.49.5 Attributes

None.

3.1.4.49.6 Groups

None.

3.1.4.49.7 Attribute Groups

None.

3.1.4.50 SetAnimation

The **SetAnimation** operation SHOULD [<96>](#) be used to apply an animation to a shape in a presentation.

The following is the WSDL port type specification of the **SetAnimation** WSDL operation.

```
<wsdl:operation name="SetAnimation" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetAnimation"
    message="tns4:IPptEdit_SetAnimation_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetAnimationResponse"
    message="tns4:IPptEdit_SetAnimation_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_SetAnimation_InputMessage** request message, and the protocol server responds with an **IPptEdit_SetAnimation_OutputMessage** response message.

3.1.4.50.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetAnimation_InputMessage	The request WSDL message for the SetAnimation WSDL operation.
IPptEdit_SetAnimation_OutputMessage	The response WSDL message for the SetAnimation WSDL operation.

3.1.4.50.1.1 IPptEdit_SetAnimation_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetAnimation
```

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

3.1.4.50.1.2 IPptEdit_SetAnimation_OutputMessage

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

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

3.1.4.50.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetAnimation	The input data for the SetAnimation WSDL operation.
SetAnimationResponse	The result data for the SetAnimation WSDL operation.

3.1.4.50.2.1 SetAnimation

The **SetAnimation** element specifies the input data for the **SetAnimation** WSDL operation.

```
<xs:element name="SetAnimation" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="fxId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="direction" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="sequenceId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the shape.

fxId: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the animation. The value **MUST** be a value from the following table.

Value	Meaning
0x0	None
0x1	Appear
0x2	Fly In
0x3	Fade

direction: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the direction of the animation. The value MUST be zero if **fxId** is not 2. The value MUST be a value from the following table if **fxId** is 2.

Value	Meaning
0x0	None
0x1	Up
0x2	Right
0x3	Down
0x4	Left

sequenceId: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the sequence type of the animation. The value MUST be a value from the following table.

Value	Meaning
0x0	Default
0x1	As One Object
0x2	By Paragraph
0x3	One by One

3.1.4.50.2.2 SetAnimationResponse

The **SetAnimationResponse** element specifies the result data for the **SetAnimation** WSDL operation.

```
<xs:element name="SetAnimationResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetAnimationResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SetAnimationResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.50.3 Complex Types

None.

3.1.4.50.4 Simple Types

None.

3.1.4.50.5 Attributes

None.

3.1.4.50.6 Groups

None.

3.1.4.50.7 Attribute Groups

None.

3.1.4.51 SetShapeAlignment

The **SetShapeAlignment** operation is used to set the alignment of a shape.

The following is the WSDL port type specification of the **SetShapeAlignment** WSDL operation.

```
<wsdl:operation name="SetShapeAlignment" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetShapeAlignment"
    message="tns4:IPptEdit_SetShapeAlignment_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetShapeAlignmentResponse"
    message="tns4:IPptEdit_SetShapeAlignment_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_SetShapeAlignment_InputMessage** request message, and the protocol server responds with an **IPptEdit_SetShapeAlignment_OutputMessage** response message.

3.1.4.51.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetShapeAlignment_InputMessage	The request WSDL message for the SetShapeAlignment WSDL operation.
IPptEdit_SetShapeAlignment_OutputMessage	The response WSDL message for the SetShapeAlignment WSDL operation.

3.1.4.51.1.1 IPptEdit_SetShapeAlignment_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetShapeAlignment
```

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

3.1.4.51.1.2 IPptEdit_SetShapeAlignment_OutputMessage

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

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

3.1.4.51.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetShapeAlignment	The input data for the SetShapeAlignment WSDL operation.
SetShapeAlignmentResponse	The result data for the SetShapeAlignment WSDL operation.

3.1.4.51.2.1 SetShapeAlignment

The **SetShapeAlignment** element specifies the input data for the **SetShapeAlignment** WSDL operation.

```
<xs:element name="SetShapeAlignment" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="textAlignment" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the shape.

textAlignment: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the alignment type. The value **MUST** be a value from the following table.

Value	Meaning
0x0	Center
0x1	Left
0x2	Right
0x3	Justify

3.1.4.51.2.2 SetShapeAlignmentResponse

The **SetShapeAlignmentResponse** element specifies the result data for the **SetShapeAlignment** WSDL operation.

```
<xs:element name="SetShapeAlignmentResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeAlignmentResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SetShapeAlignmentResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.51.3 Complex Types

None.

3.1.4.51.4 Simple Types

None.

3.1.4.51.5 Attributes

None.

3.1.4.51.6 Groups

None.

3.1.4.51.7 Attribute Groups

None.

3.1.4.52 SetShapeBold

The **SetShapeBold** operation is used to make the text in a shape bold.

The following is the WSDL port type specification of the **SetShapeBold** WSDL operation.

```
<wsdl:operation name="SetShapeBold" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeBold"
message="tns4:IPptEdit_SetShapeBold_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeBoldResponse"
message="tns4:IPptEdit_SetShapeBold_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_SetShapeBold_InputMessage** request message, and the protocol server responds with an **IPptEdit_SetShapeBold_OutputMessage** response message.

3.1.4.52.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetShapeBold_InputMessage	The request WSDL message for the SetShapeBold WSDL operation.
IPptEdit_SetShapeBold_OutputMessage	The response WSDL message for the SetShapeBold WSDL operation.

3.1.4.52.1.1 IPptEdit_SetShapeBold_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetShapeBold
```

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

3.1.4.52.1.2 IPptEdit_SetShapeBold_OutputMessage

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

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

3.1.4.52.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetShapeBold	The input data for the SetShapeBold WSDL operation.
SetShapeBoldResponse	The result data for the SetShapeBold WSDL operation.

3.1.4.52.2.1 SetShapeBold

The **SetShapeBold** element specifies the input data for the **SetShapeBold** WSDL operation.

```
<xs:element name="SetShapeBold" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="isBold" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape.

isBold: A **boolean** that specifies if the text should be made bold.

3.1.4.52.2 SetShapeBoldResponse

The **SetShapeBoldResponse** element specifies the result data for the **SetShapeBold** WSDL operation.

```
<xs:element name="SetShapeBoldResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeBoldResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SetShapeBoldResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.52.3 Complex Types

None.

3.1.4.52.4 Simple Types

None.

3.1.4.52.5 Attributes

None.

3.1.4.52.6 Groups

None.

3.1.4.52.7 Attribute Groups

None.

3.1.4.53 SetShapeBullet

The **SetShapeBullet** operation is used to make the text in a shape into a bulleted list.

The following is the WSDL port type specification of the **SetShapeBullet** WSDL operation.

```
<wsdl:operation name="SetShapeBullet" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeBullet"
```

```

message="tns4:IPptEdit_SetShapeBullet_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeBulletResponse"
message="tns4:IPptEdit_SetShapeBullet_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptEdit_SetShapeBullet_InputMessage** request message, and the protocol server responds with an **IPptEdit_SetShapeBullet_OutputMessage** response message.

3.1.4.53.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetShapeBullet_InputMessage	The request WSDL message for the SetShapeBullet WSDL operation.
IPptEdit_SetShapeBullet_OutputMessage	The response WSDL message for the SetShapeBullet WSDL operation.

3.1.4.53.1.1 IPptEdit_SetShapeBullet_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetShapeBullet

```

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

3.1.4.53.1.2 IPptEdit_SetShapeBullet_OutputMessage

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

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

3.1.4.53.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetShapeBullet	The input data for the SetShapeBullet WSDL operation.
SetShapeBulletResponse	The result data for the SetShapeBullet WSDL operation.

3.1.4.53.2.1 SetShapeBullet

The **SetShapeBullet** element specifies the input data for the **SetShapeBullet** WSDL operation.

```
<xs:element name="SetShapeBullet" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="textBullet" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the shape.

textBullet: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the bullet type. The value **MUST** be a value from the following table.

Value	Meaning
0x0	None
0x1	Bullet
0x2	Number

3.1.4.53.2.2 SetShapeBulletResponse

The **SetShapeBulletResponse** element specifies the result data for the **SetShapeBullet** WSDL operation.

```
<xs:element name="SetShapeBulletResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeBulletResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SetShapeBulletResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.53.3 Complex Types

None.

3.1.4.53.4 Simple Types

None.

3.1.4.53.5 Attributes

None.

3.1.4.53.6 Groups

None.

3.1.4.53.7 Attribute Groups

None.

3.1.4.54 SetShapeFontColor

The **SetShapeFontColor** operation is used to set the font color of a shape.

The following is the WSDL port type specification of the **SetShapeFontColor** WSDL operation.

```
<wsdl:operation name="SetShapeFontColor" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetShapeFontColor"
    message="tns4:IPptEdit_SetShapeFontColor_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetShapeFontColorResponse"
    message="tns4:IPptEdit_SetShapeFontColor_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_SetShapeFontColor_InputMessage** request message, and the protocol server responds with an **IPptEdit_SetShapeFontColor_OutputMessage** response message.

3.1.4.54.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetShapeFontColor_InputMessage	The request WSDL message for the SetShapeFontColor WSDL operation.
IPptEdit_SetShapeFontColor_OutputMessage	The response WSDL message for the SetShapeFontColor WSDL operation.

3.1.4.54.1.1 IPptEdit_SetShapeFontColor_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetShapeFontColor
```

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

3.1.4.54.1.2 IPptEdit_SetShapeFontColor_OutputMessage

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

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

3.1.4.54.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetShapeFontColor	The input data for the SetShapeFontColor WSDL operation.
SetShapeFontColorResponse	The result data for the SetShapeFontColor WSDL operation.

3.1.4.54.2.1 SetShapeFontColor

The **SetShapeFontColor** element specifies the input data for the **SetShapeFontColor** WSDL operation.

```
<xs:element name="SetShapeFontColor" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="RGBColor" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the presentation containing the shape, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([ISO/IEC29500-1:2011] section 19.7.13) that specifies the identifier of the presentation slide containing the shape.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape.

RGBColor: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the red, green, and blue components of the color.

3.1.4.54.2.2 SetShapeFontColorResponse

The **SetShapeFontColorResponse** element specifies the result data for the **SetShapeFontColor** WSDL operation.

```
<xs:element name="SetShapeFontColorResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeFontColorResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
```

</xs:element>

SetShapeFontColorResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.54.3 Complex Types

None.

3.1.4.54.4 Simple Types

None.

3.1.4.54.5 Attributes

None.

3.1.4.54.6 Groups

None.

3.1.4.54.7 Attribute Groups

None.

3.1.4.55 SetShapeFontName

The **SetShapeFontName** operation is used to set the font name for text for a shape.

The following is the WSDL port type specification of the **SetShapeFontName** WSDL operation.

```
<wsdl:operation name="SetShapeFontName" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetShapeFontName"
    message="tns4:IPptEdit_SetShapeFontName_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetShapeFontNameResponse"
    message="tns4:IPptEdit_SetShapeFontName_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_SetShapeFontName_InputMessage** request message, and the protocol server responds with an **IPptEdit_SetShapeFontName_OutputMessage** response message.

3.1.4.55.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetShapeFontName_InputMessage	The request WSDL message for the SetShapeFontName WSDL operation.

Message	Description
IPptEdit_SetShapeFontName_OutputMessage	The response WSDL message for the SetShapeFontName WSDL operation.

3.1.4.55.1.1 IPptEdit_SetShapeFontName_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetShapeFontName
```

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

3.1.4.55.1.2 IPptEdit_SetShapeFontName_OutputMessage

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

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

3.1.4.55.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetShapeFontName	The input data for the SetShapeFontName WSDL operation.
SetShapeFontNameResponse	The result data for the SetShapeFontName WSDL operation.

3.1.4.55.2.1 SetShapeFontName

The **SetShapeFontName** element specifies the input data for the **SetShapeFontName** WSDL operation.

```
<xs:element name="SetShapeFontName" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="fontName" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape.

fontName: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the font name.

3.1.4.55.2 SetShapeFontNameResponse

The **SetShapeFontNameResponse** element specifies the result data for the **SetShapeFontName** WSDL operation.

```
<xs:element name="SetShapeFontNameResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeFontNameResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SetShapeFontNameResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.55.3 Complex Types

None.

3.1.4.55.4 Simple Types

None.

3.1.4.55.5 Attributes

None.

3.1.4.55.6 Groups

None.

3.1.4.55.7 Attribute Groups

None.

3.1.4.56 SetShapeFontSize

The **SetShapeFontSize** operation is used to set the font size for text for a shape.

The following is the WSDL port type specification of the **SetShapeFontSize** WSDL operation.

```
<wsdl:operation name="SetShapeFontSize" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeFontSize"
message="tns4:IPptEdit_SetShapeFontSize_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeFontSizeResponse"
message="tns4:IPptEdit_SetShapeFontSize_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
```

</wsdl:operation>

The protocol client sends an **IPptEdit_SetShapeFontSize_InputMessage** request message, and the protocol server responds with an **IPptEdit_SetShapeFontSize_OutputMessage** response message.

3.1.4.56.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetShapeFontSize_InputMessage	The request WSDL message for the SetShapeFontSize WSDL operation.
IPptEdit_SetShapeFontSize_OutputMessage	The response WSDL message for the SetShapeFontSize WSDL operation.

3.1.4.56.1.1 IPptEdit_SetShapeFontSize_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetShapeFontSize
```

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

3.1.4.56.1.2 IPptEdit_SetShapeFontSize_OutputMessage

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

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

3.1.4.56.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetShapeFontSize	The input data for the SetShapeFontSize WSDL operation.
SetShapeFontSizeResponse	The result data for the SetShapeFontSize WSDL operation.

3.1.4.56.2.1 SetShapeFontSize

The **SetShapeFontSize** element specifies the input data for the **SetShapeFontSize** WSDL operation.

```
<xs:element name="SetShapeFontSize" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>
```

```

<xs:sequence>
  <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
  <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
  <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
  <xs:element minOccurs="0" name="fontSize" type="xs:unsignedInt"/>
</xs:sequence>
</xs:complexType>
</xs:element>

```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape.

shapeId: An **ST_DrawingElementId** ([\[ISO/IEC29500-1:2011\]](#) section 20.1.10.21) that specifies the identifier of the shape.

fontSize: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the font size.

3.1.4.56.2.2 SetShapeFontSizeResponse

The **SetShapeFontSizeResponse** element specifies the result data for the **SetShapeFontSize** WSDL operation.

```

<xs:element name="SetShapeFontSizeResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeFontSizeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

SetShapeFontSizeResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.56.3 Complex Types

None.

3.1.4.56.4 Simple Types

None.

3.1.4.56.5 Attributes

None.

3.1.4.56.6 Groups

None.

3.1.4.56.7 Attribute Groups

None.

3.1.4.57 SetShapeItalic

The **SetShapeItalic** operation is used to italicize the text for a shape.

The following is the WSDL port type specification of the **SetShapeItalic** WSDL operation.

```
<wsdl:operation name="SetShapeItalic" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetShapeItalic"
    message="tns4:IPptEdit_SetShapeItalic_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetShapeItalicResponse"
    message="tns4:IPptEdit_SetShapeItalic_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_SetShapeItalic_InputMessage** request message, and the protocol server responds with an **IPptEdit_SetShapeItalic_OutputMessage** response message.

3.1.4.57.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetShapeItalic_InputMessage	The request WSDL message for the SetShapeItalic WSDL operation.
IPptEdit_SetShapeItalic_OutputMessage	The response WSDL message for the SetShapeItalic WSDL operation.

3.1.4.57.1.1 IPptEdit_SetShapeItalic_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetShapeItalic
```

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

3.1.4.57.1.2 IPptEdit_SetShapeItalic_OutputMessage

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

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

3.1.4.57.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetShapeItalic	The input data for the SetShapeItalic WSDL operation.
SetShapeItalicResponse	The result data for the SetShapeItalic WSDL operation.

3.1.4.57.2.1 SetShapeItalic

The **SetShapeItalic** element specifies the input data for the **SetShapeItalic** WSDL operation.

```
<xs:element name="SetShapeItalic" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="isItalic" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape.

isItalic: A **boolean** that specifies if the text should be italicized.

3.1.4.57.2.2 SetShapeItalicResponse

The **SetShapeItalicResponse** element specifies the result data for the **SetShapeItalic** WSDL operation.

```
<xs:element name="SetShapeItalicResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeItalicResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SetShapeItalicResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.57.3 Complex Types

None.

3.1.4.57.4 Simple Types

None.

3.1.4.57.5 Attributes

None.

3.1.4.57.6 Groups

None.

3.1.4.57.7 Attribute Groups

None.

3.1.4.58 SetShapeTextDirection

The **SetShapeTextDirection** operation is used to set the text direction for a shape.

The following is the WSDL port type specification of the **SetShapeTextDirection** WSDL operation.

```
<wsdl:operation name="SetShapeTextDirection" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetShapeTextDirection"
    message="tns4:IPptEdit_SetShapeTextDirection_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetShapeTextDirectionResponse"
    message="tns4:IPptEdit_SetShapeTextDirection_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_SetShapeTextDirection_InputMessage** request message, and the protocol server responds with an **IPptEdit_SetShapeTextDirection_OutputMessage** response message.

3.1.4.58.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetShapeTextDirection_InputMessage	The request WSDL message for the SetShapeTextDirection WSDL operation.
IPptEdit_SetShapeTextDirection_OutputMessage	The response WSDL message for the SetShapeTextDirection WSDL operation.

3.1.4.58.1.1 IPptEdit_SetShapeTextDirection_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetShapeTextDirection
```

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

3.1.4.58.1.2 IPptEdit_SetShapeTextDirection_OutputMessage

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

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

3.1.4.58.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetShapeTextDirection	The input data for the SetShapeTextDirection WSDL operation.
SetShapeTextDirectionResponse	The result data for the SetShapeTextDirection WSDL operation.

3.1.4.58.2.1 SetShapeTextDirection

The **SetShapeTextDirection** element specifies the input data for the **SetShapeTextDirection** WSDL operation.

```
<xs:element name="SetShapeTextDirection" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="textDirection" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([XMLSCHEMA2] section 3.2.1) that specifies the identifier of the presentation containing the shape, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([ISO/IEC29500-1:2011] section 19.7.13) that specifies the identifier of the presentation slide containing the shape.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape.

textDirection: An **unsignedInt** ([XMLSCHEMA2] section 3.3.22) that specifies the text direction. The value **MUST** be a value from the following table.

Value	Meaning
0x0	Neutral
0x1	Left to Right
0x2	Right to Left

3.1.4.58.2.2 SetShapeTextDirectionResponse

The **SetShapeTextDirectionResponse** element specifies the result data for the **SetShapeTextDirection** WSDL operation.

```
<xs:element name="SetShapeTextDirectionResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeTextDirectionResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SetShapeTextDirectionResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.58.3 Complex Types

None.

3.1.4.58.4 Simple Types

None.

3.1.4.58.5 Attributes

None.

3.1.4.58.6 Groups

None.

3.1.4.58.7 Attribute Groups

None.

3.1.4.59 SetShapeUnderline

The **SetShapeUnderline** operation is used to underline text in a shape.

The following is the WSDL port type specification of the **SetShapeUnderline** WSDL operation.

```
<wsdl:operation name="SetShapeUnderline" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeUnderline"
message="tns4:IPptEdit_SetShapeUnderline_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeUnderlineResponse"
message="tns4:IPptEdit_SetShapeUnderline_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_SetShapeUnderline_InputMessage** request message, and the protocol server responds with an **IPptEdit_SetShapeUnderline_OutputMessage** response message.

3.1.4.59.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetShapeUnderline_InputMessage	The request WSDL message for the SetShapeUnderline WSDL operation.
IPptEdit_SetShapeUnderline_OutputMessage	The response WSDL message for the SetShapeUnderline WSDL operation.

3.1.4.59.1.1 IPptEdit_SetShapeUnderline_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetShapeUnderline
```

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

3.1.4.59.1.2 IPptEdit_SetShapeUnderline_OutputMessage

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

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

3.1.4.59.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetShapeUnderline	The input data for the SetShapeUnderline WSDL operation.
SetShapeUnderlineResponse	The result data for the SetShapeUnderline WSDL operation.

3.1.4.59.2.1 SetShapeUnderline

The **SetShapeUnderline** element specifies the input data for the **SetShapeUnderline** WSDL operation.

```
<xs:element name="SetShapeUnderline" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="isUnderline" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
```

</xs:element>

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape.

isUnderline: A **boolean** that specifies if the shape text should be underlined.

3.1.4.59.2.2 SetShapeUnderlineResponse

The **SetShapeUnderlineResponse** element specifies the result data for the **SetShapeUnderline** WSDL operation.

```
<xs:element name="SetShapeUnderlineResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeUnderlineResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SetShapeUnderlineResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.59.3 Complex Types

None.

3.1.4.59.4 Simple Types

None.

3.1.4.59.5 Attributes

None.

3.1.4.59.6 Groups

None.

3.1.4.59.7 Attribute Groups

None.

3.1.4.60 SetTransition

The **SetTransition** operation SHOULD [<97>](#) be used to apply a transition to a slide in a presentation.

The following is the WSDL port type specification of the **SetTransition** WSDL operation.

```

<wsdl:operation name="SetTransition" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetTransition"
    message="tns4:IPptEdit_SetTransition_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/SetTransitionResponse"
    message="tns4:IPptEdit_SetTransition_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IPptEdit_SetTransition_InputMessage** request message, and the protocol server responds with an **IPptEdit_SetTransition_OutputMessage** response message.

3.1.4.60.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetTransition_InputMessage	The request WSDL message for the SetTransition WSDL operation.
IPptEdit_SetTransition_OutputMessage	The response WSDL message for the SetTransition WSDL operation.

3.1.4.60.1.1 IPptEdit_SetTransition_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetTransition

```

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

3.1.4.60.1.2 IPptEdit_SetTransition_OutputMessage

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

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

3.1.4.60.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetTransition	The input data for the SetTransition WSDL operation.
SetTransitionResponse	The result data for the SetTransition WSDL operation.

3.1.4.60.2.1 SetTransition

The **SetTransition** element specifies the input data for the **SetTransition** WSDL operation.

```
<xs:element name="SetTransition" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="type" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="direction" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="duration" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="isManualAdvance" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the slide, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide.

type: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the type of transition. The value **MUST** be a value from the following table.

Value	Meaning
0x0	None
0x6	Fade through black
0x17	Fade smoothly
0x14	Push

direction: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the direction of transition. If **type** is not 0x14, the value **MUST** be zero. If **type** is 0x14, the value **MUST** be a value from the following table.

Value	Meaning
0x0	From right
0x1	From bottom
0x2	From left
0x3	From top

duration: An **unsignedInt** ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the duration of the transition in milliseconds.

isManualAdvance: Reserved. **MUST** be ignored.

3.1.4.60.2.2 SetTransitionResponse

The **SetTransitionResponse** element specifies the result data for the **SetTransition** WSDL operation.

```
<xs:element name="SetTransitionResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetTransitionResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SetTransitionResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.60.3 Complex Types

None.

3.1.4.60.4 Simple Types

None.

3.1.4.60.5 Attributes

None.

3.1.4.60.6 Groups

None.

3.1.4.60.7 Attribute Groups

None.

3.1.4.61 SetTransitionApplyAll

The **SetTransitionApplyAll** operation SHOULD [<98>](#) be used to copy a transition from one slide and apply it to all other slides in a presentation.

The following is the WSDL port type specification of the **SetTransitionApplyAll** WSDL operation.

```
<wsdl:operation name="SetTransitionApplyAll" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input>
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetTransitionApplyAll"
    message="tns4:IPptEdit_SetTransitionApplyAll_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output>
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetTransitionApplyAllResponse"
    message="tns4:IPptEdit_SetTransitionApplyAll_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_SetTransitionApplyAll_InputMessage** request message, and the protocol server responds with an **IPptEdit_SetTransitionApplyAll_OutputMessage** response message.

3.1.4.61.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetTransitionApplyAll_InputMessage	The request WSDL message for the SetTransitionApplyAll WSDL operation.
IPptEdit_SetTransitionApplyAll_OutputMessage	The response WSDL message for the SetTransitionApplyAll WSDL operation.

3.1.4.61.1.1 IPptEdit_SetTransitionApplyAll_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetTransitionApplyAll
```

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

3.1.4.61.1.2 IPptEdit_SetTransitionApplyAll_OutputMessage

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

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

3.1.4.61.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetTransitionApplyAll	The input data for the SetTransitionApplyAll WSDL operation.
SetTransitionApplyAllResponse	The result data for the SetTransitionApplyAll WSDL operation.

3.1.4.61.2.1 SetTransitionApplyAll

The **SetTransitionApplyAll** element specifies the input data for the **SetTransitionApplyAll** WSDL operation.

```
<xs:element name="SetTransitionApplyAll" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

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

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the transition to be copied.

3.1.4.61.2.2 SetTransitionApplyAllResponse

The **SetTransitionApplyAllResponse** element specifies the result data for the **SetTransitionApplyAll** WSDL operation.

```
<xs:element name="SetTransitionApplyAllResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetTransitionApplyAllResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SetTransitionApplyAllResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element **MUST** be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.61.3 Complex Types

None.

3.1.4.61.4 Simple Types

None.

3.1.4.61.5 Attributes

None.

3.1.4.61.6 Groups

None.

3.1.4.61.7 Attribute Groups

None.

3.1.4.62 SetWordArt

Reserved. **MUST** not be called.

The following is the WSDL port type specification of the **SetWordArt** WSDL operation.

```
<wsdl:operation name="SetWordArt" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
```

```

ServerInternalService/IPptEdit/SetWordArt" message="tns4:IPptEdit_SetWordArt_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetWordArtResponse"
message="tns4:IPptEdit_SetWordArt_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

Reserved. MUST not be called.

3.1.4.62.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_SetWordArt_InputMessage	The request WSDL message for the SetWordArt WSDL operation.
IPptEdit_SetWordArt_OutputMessage	The response WSDL message for the SetWordArt WSDL operation.

3.1.4.62.1.1 IPptEdit_SetWordArt_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/SetWordArt

```

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

3.1.4.62.1.2 IPptEdit_SetWordArt_OutputMessage

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

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

3.1.4.62.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
SetWordArt	The input data for the SetWordArt WSDL operation.
SetWordArtResponse	The result data for the SetWordArt WSDL operation.

3.1.4.62.2.1 SetWordArt

The **SetWordArt** element specifies the input data for the **SetWordArt** WSDL operation.

```

<xs:element name="SetWordArt" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="styleId" type="xs:int"/>
      <xs:element minOccurs="0" name="isClear" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

presentationId: Reserved. MUST be ignored.

slideId: Reserved. MUST be ignored.

shapeId: Reserved. MUST be ignored.

styleId: Reserved. MUST be ignored.

isClear: Reserved. MUST be ignored.

3.1.4.62.2 SetWordArtResponse

The **SetWordArtResponse** element specifies the result data for the **SetWordArt** WSDL operation.

```

<xs:element name="SetWordArtResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetWordArtResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

SetWordArtResult: Reserved. MUST be ignored.

3.1.4.62.3 Complex Types

None.

3.1.4.62.4 Simple Types

None.

3.1.4.62.5 Attributes

None.

3.1.4.62.6 Groups

None.

3.1.4.62.7 Attribute Groups

None.

3.1.4.63 ShapeFormatPainting

The **ShapeFormatPainting** operation SHOULD [<99>](#) be used to copy formatting from one shape and apply it to another shape in the presentation.

The following is the WSDL port type specification of the **ShapeFormatPainting** WSDL operation.

```
<wsdl:operation name="ShapeFormatPainting" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ShapeFormatPainting"
    message="tns4:IPptEdit_ShapeFormatPainting_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ShapeFormatPaintingResponse"
    message="tns4:IPptEdit_ShapeFormatPainting_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ShapeFormatPainting_InputMessage** request message, and the protocol server responds with an **IPptEdit_ShapeFormatPainting_OutputMessage** response message.

3.1.4.63.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ShapeFormatPainting_InputMessage	The request WSDL message for the ShapeFormatPainting WSDL operation.
IPptEdit_ShapeFormatPainting_OutputMessage	The response WSDL message for the ShapeFormatPainting WSDL operation.

3.1.4.63.1.1 IPptEdit_ShapeFormatPainting_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ShapeFormatPainting
```

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

3.1.4.63.1.2 IPptEdit_ShapeFormatPainting_OutputMessage

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

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

3.1.4.63.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ShapeFormatPainting	The input data for the ShapeFormatPainting WSDL operation.
ShapeFormatPaintingResponse	The result data for the ShapeFormatPainting WSDL operation.

3.1.4.63.2.1 ShapeFormatPainting

The **ShapeFormatPainting** element specifies the input data for the **ShapeFormatPainting** WSDL operation.

```
<xs:element name="ShapeFormatPainting" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="sourceSlideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="sourceShapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="destinationSlideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="destinationShapeId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

sourceSlideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that the formatting is copied from.

sourceShapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape that the formatting is copied from. The shape **MUST NOT** be a group, SmartArt, table, chart, or ActiveX shape.

destinationSlideId: An **ST_SlideId** ([ISO/IEC29500-1:2011] section 19.7.13) that specifies the identifier of the presentation slide containing the shape that the formatting is applied to.

destinationShapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape that the formatting is applied to. The shape **MUST NOT** be a SmartArt, table, chart, or ActiveX shape.

3.1.4.63.2.2 ShapeFormatPaintingResponse

The **ShapeFormatPaintingResponse** element specifies the result data for the **ShapeFormatPainting** WSDL operation.

```
<xs:element name="ShapeFormatPaintingResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ShapeFormatPaintingResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```


ShapeFormatPaintingResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.63.3 Complex Types

None.

3.1.4.63.4 Simple Types

None.

3.1.4.63.5 Attributes

None.

3.1.4.63.6 Groups

None.

3.1.4.63.7 Attribute Groups

None.

3.1.4.64 ShowHideSlide

The **ShowHideSlide** operation is used to show or hide a presentation slide.

The following is the WSDL port type specification of the **ShowHideSlide** WSDL operation.

```
<wsdl:operation name="ShowHideSlide" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ShowHideSlide"
    message="tns4:IPptEdit_ShowHideSlide_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
    ServerInternalService/IPptEdit/ShowHideSlideResponse"
    message="tns4:IPptEdit_ShowHideSlide_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_ShowHideSlide_InputMessage** request message, and the protocol server responds with an **IPptEdit_ShowHideSlide_OutputMessage** response message.

3.1.4.64.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_ShowHideSlide_InputMessage	The request WSDL message for the ShowHideSlide WSDL operation.
IPptEdit_ShowHideSlide_OutputMessage	The response WSDL message for the ShowHideSlide WSDL operation.

3.1.4.64.1.1 IPptEdit_ShowHideSlide_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/ShowHideSlide
```

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

3.1.4.64.1.2 IPptEdit_ShowHideSlide_OutputMessage

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

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

3.1.4.64.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ShowHideSlide	The input data for the ShowHideSlide WSDL operation.
ShowHideSlideResponse	The result data for the ShowHideSlide WSDL operation.

3.1.4.64.2.1 ShowHideSlide

The **ShowHideSlide** element specifies the input data for the **ShowHideSlide** WSDL operation.

```
<xs:element name="ShowHideSlide" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="fHidden" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the target presentation slide, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the target presentation.

fHidden: A **boolean** ([\[XMLSCHEMA2\]](#) section 3.2.2) that specifies if the operation is to show or hide the presentation slide. If the value is true, the protocol server **MUST** make the presentation slide hidden. Otherwise, the protocol server **MUST** make the presentation slide shown. **MUST** be present.

3.1.4.64.2.2 ShowHideSlideResponse

The **ShowHideSlideResponse** element specifies the result data for the **ShowHideSlide** WSDL operation.

```
<xs:element name="ShowHideSlideResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ShowHideSlideResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ShowHideSlideResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.64.3 Complex Types

None.

3.1.4.64.4 Simple Types

None.

3.1.4.64.5 Attributes

None.

3.1.4.64.6 Groups

None.

3.1.4.64.7 Attribute Groups

None.

3.1.4.65 Undo

The **Undo** operation is used to undo the last operation in a presentation that is not undone.

The following is the WSDL port type specification of the **Undo** WSDL operation.

```
<wsdl:operation name="Undo" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/Undo" message="tns4:IPptEdit Undo InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/UndoResponse" message="tns4:IPptEdit Undo OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_Undo_InputMessage** request message, and the protocol server responds with an **IPptEdit_Undo_OutputMessage** response message.

3.1.4.65.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_Undo_InputMessage	The request WSDL message for the Undo WSDL operation.
IPptEdit_Undo_OutputMessage	The response WSDL message for the Undo WSDL operation.

3.1.4.65.1.1 IPptEdit_Undo_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/Undo
```

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

3.1.4.65.1.2 IPptEdit_Undo_OutputMessage

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

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

3.1.4.65.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
Undo	The input data for the Undo WSDL operation.
UndoResponse	The result data for the Undo WSDL operation.

3.1.4.65.2.1 Undo

The **Undo** element specifies the input data for the **Undo** WSDL operation.

```
<xs:element name="Undo" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

presentationId: A **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation, as specified in section [3.1.4.27](#).

3.1.4.65.2.2 UndoResponse

The **UndoResponse** element specifies the result data for the **Undo** WSDL operation.

```
<xs:element name="UndoResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="UndoResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

UndoResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.65.3 Complex Types

None.

3.1.4.65.4 Simple Types

None.

3.1.4.65.5 Attributes

None.

3.1.4.65.6 Groups

None.

3.1.4.65.7 Attribute Groups

None.

3.1.4.66 UngroupShape

The **UngroupShape** operation SHOULD [<100>](#) be used to ungroup a shape in a presentation.

The following is the WSDL port type specification of the **UngroupShape** WSDL operation.

```
<wsdl:operation name="UngroupShape" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/UngroupShape"
message="tns4:IPptEdit_UngroupShape_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/UngroupShapeResponse"
message="tns4:IPptEdit_UngroupShape_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IPptEdit_UngroupShape_InputMessage** request message, and the protocol server responds with an **IPptEdit_UngroupShape_OutputMessage** response message.

3.1.4.66.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_UngroupShape_InputMessage	The request WSDL message for the UngroupShape WSDL operation.
IPptEdit_UngroupShape_OutputMessage	The response WSDL message for the UngroupShape WSDL operation.

3.1.4.66.1.1 IPptEdit_UngroupShape_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/UngroupShape
```

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

3.1.4.66.1.2 IPptEdit_UngroupShape_OutputMessage

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

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

3.1.4.66.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
UngroupShape	The input data for the UngroupShape WSDL operation.
UngroupShapeResponse	The result data for the UngroupShape WSDL operation.

3.1.4.66.2.1 UngroupShape

The **UngroupShape** element specifies the input data for the **UngroupShape** WSDL operation.

```
<xs:element name="UngroupShape" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

presentationId: A string ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the presentation containing the shape that is being modified, as specified in section [3.1.4.27](#).

slideId: An **ST_SlideId** ([\[ISO/IEC29500-1:2011\]](#) section 19.7.13) that specifies the identifier of the presentation slide containing the shape that is being modified.

shapeId: An **ST_DrawingElementId** ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape that is being modified.

3.1.4.66.2 UngroupShapeResponse

The **UngroupShapeResponse** element specifies the result data for the **UngroupShape** WSDL operation.

```
<xs:element name="UngroupShapeResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="UngroupShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

UngroupShapeResult: A **ServiceResult** (section [2.2.4.43](#)) that specifies the result of the operation. The **Result** child element MUST be an **EditCommandResponse** (section [2.2.4.23](#)).

3.1.4.66.3 Complex Types

None.

3.1.4.66.4 Simple Types

None.

3.1.4.66.5 Attributes

None.

3.1.4.66.6 Groups

None.

3.1.4.66.7 Attribute Groups

None.

3.1.4.67 UpdateCanary

Reserved. MUST be ignored.

The following is the WSDL port type specification of the **UpdateCanary** WSDL operation.

```
<wsdl:operation name="UpdateCanary" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/UpdateCanary"
message="tns4:IPptEdit_UpdateCanary_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/UpdateCanaryResponse"
```

```

message="tns4:IPptEdit_UpdateCanary_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

3.1.4.67.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
IPptEdit_UpdateCanary_InputMessage	The request WSDL message for the UpdateCanary WSDL operation.
IPptEdit_UpdateCanary_OutputMessage	The response WSDL message for the UpdateCanary WSDL operation.

3.1.4.67.1.1 IPptEdit_UpdateCanary_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/IPptEdit/UpdateCanary

```

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

3.1.4.67.1.2 IPptEdit_UpdateCanary_OutputMessage

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

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

3.1.4.67.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
UpdateCanary	The input data for the UpdateCanary WSDL operation.
UpdateCanaryResponse	The result data for the UpdateCanary WSDL operation.

3.1.4.67.2.1 UpdateCanary

The **UpdateCanary** element specifies the input data for the **UpdateCanary** WSDL operation.

```

<xs:element name="UpdateCanary" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>

```



```
</xs:element>
```

presentationId: Reserved. MUST be ignored.

3.1.4.67.2.2 UpdateCanaryResponse

The **UpdateCanaryResponse** element specifies the result data for the **UpdateCanary** WSDL operation.

```
<xs:element name="UpdateCanaryResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="UpdateCanaryResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

UpdateCanaryResult: Reserved. MUST be ignored.

3.1.4.67.3 Complex Types

None.

3.1.4.67.4 Simple Types

None.

3.1.4.67.5 Attributes

None.

3.1.4.67.6 Groups

None.

3.1.4.67.7 Attribute Groups

None.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

4 Protocol Examples

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

4.1 Sample Protocol Interaction

In this example, the protocol client sends a request to the protocol server for the identifier of a presentation. The following **IPptEdit_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>http://protocolserver/Shared
      Documents/MSPWEDPSExample.pptx</rawDocId></GetPresentationId>
    </s:Body>
  </s:Envelope>
```

The protocol server responds with the appropriate identifier for the presentation. The following **IPptEdit_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
    m568ced57b5cc4aeda74fff96066a8267m</GetPresentationIdResult></GetPresentationIdResponse>
  </s:Body>
</s:Envelope>
```

The response sent to the protocol client also contains the **canary** which is sent back as a **cookie** with the following name: "X-Key". This canary is sent with any subsequent operations to the protocol server that can modify the presentation.

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

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body><GetEditPresInfo
      xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServer
      InternalService/">
    <presentationId>d=F39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a9a007e0723241fm568ced57
    b5cc4aeda74fff96066a8267m</presentationId>
    <firstSlideIndex>0</firstSlideIndex>
    <numSlides>10</numSlides>
    <includedEditSlideIndex>1</includedEditSlideIndex>
  </GetEditPresInfo>
  </s:Body>
</s:Envelope>
```

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

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetEditPresInfoResponse
      xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServer
      InternalService/">
      <GetEditPresInfoResult 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="a:EditPresentationInfo">
          <a:EditSlideInfos>
            <a:EditSlideInfoFe>
              <a:Hidden>false</a:Hidden>
              <a:Id>256</a:Id>
              <a:Thumb>
                <a:Alt i:nil="true"/>
                <a:Clip>
                  <a:Bottom>96</a:Bottom>
                  <a:Left>0</a:Left>
                  <a:Right>128</a:Right>
                  <a:Top>0</a:Top>
                </a:Clip>
                <a:Html i:nil="true"/>
                <a:Id>1</a:Id>
              </a:Thumb>
              <a:Title>Fancy Title</a:Title>
            </a:EditSlideInfoFe>
          </a:EditSlideInfos>
          <a:FirstSlideIndex>0</a:FirstSlideIndex>
          <a:HasComments>false</a:HasComments>
          <a:MainMasters>
            <a:MainMasterFe>
              <a:ContentMasters>
                <a:ContentMasterFe>
                  <a:Id>2147483649</a:Id>
                  <a:Name>Title Slide</a:Name>
                  <a:SlideLayoutType>title</a:SlideLayoutType>
                </a:ContentMasterFe>
                <a:ContentMasterFe>
                  <a:Id>2147483650</a:Id>
                  <a:Name>Title and Content</a:Name>
                  <a:SlideLayoutType>obj</a:SlideLayoutType>
                </a:ContentMasterFe>
                <a:ContentMasterFe>
                  <a:Id>2147483651</a:Id>
                  <a:Name>Section Header</a:Name>
                  <a:SlideLayoutType>secHead</a:SlideLayoutType>
                </a:ContentMasterFe>
                <a:ContentMasterFe>
                  <a:Id>2147483652</a:Id>
                  <a:Name>Two Content</a:Name>
                  <a:SlideLayoutType>twoObj</a:SlideLayoutType>
                </a:ContentMasterFe>
                <a:ContentMasterFe>
                  <a:Id>2147483653</a:Id>
                  <a:Name>Comparison</a:Name>
                  <a:SlideLayoutType>twoTxTwoObj</a:SlideLayoutType>
                </a:ContentMasterFe>
                <a:ContentMasterFe>
                  <a:Id>2147483654</a:Id>
                  <a:Name>Title Only</a:Name>
                  <a:SlideLayoutType>titleOnly</a:SlideLayoutType>
                </a:ContentMasterFe>
              </a:ContentMasters>
            </a:MainMasterFe>
          </a:MainMasters>
        </a:Result>
      </a:EditPresInfoResult>
    </s:Body>
  </s:Envelope>

```

```

        </a:ContentMasterFe>
        <a:ContentMasterFe>
          <a:Id>2147483655</a:Id>
          <a:Name>Blank</a:Name>
          <a:SlideLayoutType>blank</a:SlideLayoutType>
        </a:ContentMasterFe>
        <a:ContentMasterFe>
          <a:Id>2147483656</a:Id>
          <a:Name>Content with Caption</a:Name>
          <a:SlideLayoutType>objTx</a:SlideLayoutType>
        </a:ContentMasterFe>
        <a:ContentMasterFe>
          <a:Id>2147483657</a:Id>
          <a:Name>Picture with Caption</a:Name>
          <a:SlideLayoutType>picTx</a:SlideLayoutType>
        </a:ContentMasterFe>
      </a:ContentMasters>
      <a:Id>2147483648</a:Id>
      <a:InContent>true</a:InContent>
      <a:Name>Office Theme</a:Name>
    </a:MainMasterFe>
  </a:MainMasters>
  <a:Redo>false</a:Redo>
  <a:Rtl>false</a:Rtl>
  <a:Size>
    <a:X>620</a:X>
    <a:Y>465</a:Y>
  </a:Size>
  <a:Slide i:nil="true"/>
  <a:SlideOrder>
    <a:SlideIds
      xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
      <b:unsignedInt>256</b:unsignedInt>
    </a:SlideIds>
  </a:SlideOrder>
  <a:Undo>false</a:Undo>
</a:Result>
</GetEditPresInfoResult></GetEditPresInfoResponse>
</s:Body>
</s:Envelope>

```

Next, the protocol client sends a request to the protocol server for information about a presentation slide. The identifier of the first slide is 256 that is specified by the first element of the **SlideOrder** element of the **EditPresentationInfo** element (section [2.2.4.26](#)). The following **IPptEdit_GetEditSlide_InputMessage** message is sent to the protocol server:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"><s:Body><GetEditSlide
xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServer
InternalService/"><presentationId>d=F39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a9a007
e0723241fm568ced57b5cc4aeda74fff96066a8267m</presentationId><slideId>256</slideId></GetEditSl
ide></s:Body></s:Envelope>

```

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

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetEditSlideResponse
      xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServer
      InternalService/">
      <GetEditSlideResult 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="a:EditSlide">

```

```

<a:ContentMasterId>2147483649</a:ContentMasterId>
<a:EditShapes>
  <a:EditShape>
    <a:Alt i:nil="true"/>
    <a:EditBounds>
      <a:Bottom>140</a:Bottom>
      <a:Left>46</a:Left>
      <a:Right>573</a:Right>
      <a:Top>41</a:Top>
    </a:EditBounds>
    <a:EditHtml>&lt;div class=defaultPHStyles style='font-size:32.0px;font-
family:Calibri;color:#7293C9;text-align:center'&gt; &lt;/div&gt;&lt;p class=nonListPara
style='direction:ltr' lang=en-US style='language:en-US' align=center&gt;&lt;font size=6
face=Calibri color="#7293C9"&gt;&lt;span style='font-variant:normal;text-
transform:uppercase'&gt;&lt;b&gt;Fancy
Title&lt;/b&gt;&lt;/span&gt;&lt;/font&gt;&lt;/p&gt;&lt;/a:EditHtml>
    <a:FontSizeMapping>
      <a:Size1>12</a:Size1>
      <a:Size2>15</a:Size2>
      <a:Size3>19</a:Size3>
      <a:Size4>21</a:Size4>
      <a:Size5>28</a:Size5>
      <a:Size6>44</a:Size6>
      <a:Size7>56</a:Size7>
    </a:FontSizeMapping>
    <a:IsPicture>>false</a:IsPicture>
    <a:IsPlaceholder>>true</a:IsPlaceholder>
    <a:IsPrompt>>false</a:IsPrompt>
    <a:IsSmartArt>>false</a:IsSmartArt>
    <a:PlaceholderType>2</a:PlaceholderType>
    <a:ShapeId>2</a:ShapeId>
    <a:ShapeViewElemId>2</a:ShapeViewElemId>
    <a:SmartArtInfo i:nil="true"/>
    <a:TextViewElemId>3</a:TextViewElemId>
  </a:EditShape>
  <a:EditShape>
    <a:Alt i:nil="true"/>
    <a:EditBounds>
      <a:Bottom>340</a:Bottom>
      <a:Left>25</a:Left>
      <a:Right>459</a:Right>
      <a:Top>222</a:Top>
    </a:EditBounds>
    <a:EditHtml>&lt;div class=defaultPHStyles style='font-size:24.0px;font-
family:Calibri;color:#1F497D;text-align:center'&gt; &lt;/div&gt;&lt;p class=nonListPara
style='direction:ltr' lang=en-US style='language:en-US' align=center&gt;&lt;font size=5
face=Calibri color="#1F497D"&gt;Boring subtitle&lt;/font&gt;&lt;/p&gt;&lt;/a:EditHtml>
    <a:FontSizeMapping>
      <a:Size1>12</a:Size1>
      <a:Size2>20</a:Size2>
      <a:Size3>24</a:Size3>
      <a:Size4>28</a:Size4>
      <a:Size5>32</a:Size5>
      <a:Size6>37</a:Size6>
      <a:Size7>56</a:Size7>
    </a:FontSizeMapping>
    <a:IsPicture>>false</a:IsPicture>
    <a:IsPlaceholder>>true</a:IsPlaceholder>
    <a:IsPrompt>>false</a:IsPrompt>
    <a:IsSmartArt>>false</a:IsSmartArt>
    <a:PlaceholderType>3</a:PlaceholderType>
    <a:ShapeId>3</a:ShapeId>
    <a:ShapeViewElemId>0</a:ShapeViewElemId>
    <a:SmartArtInfo i:nil="true"/>
    <a:TextViewElemId>4</a:TextViewElemId>
  </a:EditShape>
</a:EditShapes>
<a:Hidden>>false</a:Hidden>
<a:IsTitleContentMaster>>true</a:IsTitleContentMaster>

```

```
<a:MainMasterId>2147483648</a:MainMasterId>
<a:Notes>
  <a:Html>
    <a:Text>&lt;p class=nonListPara style='direction:ltr' lang=en-US
style='language:en-US' align=left&gt;&lt;font face=Calibri&gt;&lt;img width=1 height=10
style='visibility:hidden'&gt;&lt;/font&gt;&lt;/p&gt;</a:Text>
  </a:Html>
  <a:IsPrompt>true</a:IsPrompt>
</a:Notes>
<a:SlideId>256</a:SlideId>
<a:Theme>
  <a:Accent1>
    <a:b>189</a:b>
    <a:g>129</a:g>
    <a:r>79</a:r>
  </a:Accent1>
  <a:Accent2>
    <a:b>77</a:b>
    <a:g>80</a:g>
    <a:r>192</a:r>
  </a:Accent2>
  <a:Accent3>
    <a:b>89</a:b>
    <a:g>187</a:g>
    <a:r>155</a:r>
  </a:Accent3>
  <a:Accent4>
    <a:b>162</a:b>
    <a:g>100</a:g>
    <a:r>128</a:r>
  </a:Accent4>
  <a:Accent5>
    <a:b>198</a:b>
    <a:g>172</a:g>
    <a:r>75</a:r>
  </a:Accent5>
  <a:Accent6>
    <a:b>70</a:b>
    <a:g>150</a:g>
    <a:r>247</a:r>
  </a:Accent6>
  <a:Dark1>
    <a:b>0</a:b>
    <a:g>0</a:g>
    <a:r>0</a:r>
  </a:Dark1>
  <a:Dark2>
    <a:b>125</a:b>
    <a:g>73</a:g>
    <a:r>31</a:r>
  </a:Dark2>
  <a:Light1>
    <a:b>255</a:b>
    <a:g>255</a:g>
    <a:r>255</a:r>
  </a:Light1>
  <a:Light2>
    <a:b>225</a:b>
    <a:g>236</a:g>
    <a:r>238</a:r>
  </a:Light2>
</a:Theme>
<a:ThumbnailViewElement>
  <a:Alt i:nil="true"/>
  <a:Clip>
    <a:Bottom>96</a:Bottom>
    <a:Left>0</a:Left>
    <a:Right>128</a:Right>
    <a:Top>0</a:Top>
```

```

</a:Clip>
<a:Html i:nil="true"/>
<a:Id>1</a:Id>

<a:ImgId>./editImageHandler.ashx?Pid=d%3DF39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a
9a007e0723241fm568ced57b5cc4aeda74fff96066a8267m&amp;Rid=14tvmylk%2Ed4e%2Epng</a:ImgId>
  <a:Position>
    <a:X>0</a:X>
    <a:Y>0</a:Y>
  </a:Position>
  <a:Z>0</a:Z>
</a:ThumbnailViewElement>
<a:ViewElems>
  <a:ViewElement>
    <a:Alt/>
    <a:Clip>
      <a:Bottom>205</a:Bottom>
      <a:Left>0</a:Left>
      <a:Right>407</a:Right>
      <a:Top>0</a:Top>
    </a:Clip>
    <a:Html i:nil="true"/>
    <a:Id>2</a:Id>

<a:ImgId>./editImageHandler.ashx?Pid=d%3DF39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a
9a007e0723241fm568ced57b5cc4aeda74fff96066a8267m&amp;Rid=yvkqez1r%2Esms%2Epng</a:ImgId>
  <a:Position>
    <a:X>149</a:X>
    <a:Y>8</a:Y>
  </a:Position>
  <a:Z>1</a:Z>
</a:ViewElement>
<a:ViewElement>
  <a:Alt i:nil="true"/>
  <a:Clip>
    <a:Bottom>74</a:Bottom>
    <a:Left>407</a:Left>
    <a:Right>547</a:Right>
    <a:Top>0</a:Top>
  </a:Clip>
  <a:Html i:nil="true"/>
  <a:Id>3</a:Id>

<a:ImgId>./editImageHandler.ashx?Pid=d%3DF39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a
9a007e0723241fm568ced57b5cc4aeda74fff96066a8267m&amp;Rid=yvkqez1r%2Esms%2Epng</a:ImgId>
  <a:Position>
    <a:X>245</a:X>
    <a:Y>66</a:Y>
  </a:Position>
  <a:Z>2</a:Z>
</a:ViewElement>
<a:ViewElement>
  <a:Alt i:nil="true"/>
  <a:Clip>
    <a:Bottom>24</a:Bottom>
    <a:Left>547</a:Left>
    <a:Right>708</a:Right>
    <a:Top>0</a:Top>
  </a:Clip>
  <a:Html i:nil="true"/>
  <a:Id>4</a:Id>

<a:ImgId>./editImageHandler.ashx?Pid=d%3DF39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a
9a007e0723241fm568ced57b5cc4aeda74fff96066a8267m&amp;Rid=yvkqez1r%2Esms%2Epng</a:ImgId>
  <a:Position>
    <a:X>163</a:X>
    <a:Y>232</a:Y>
  </a:Position>
  <a:Z>5</a:Z>

```

```

        </a:ViewElement>
        <a:ViewElement>
          <a:Alt i:nil="true"/>
          <a:Clip i:nil="true"/>
          <a:Html>&lt;div
style="position:absolute;top:0px;left:0px;width:620px;height:465px;background-
color:#FFFFFF"/&gt;</a:Html>
          <a:Id>0</a:Id>
          <a:ImgId i:nil="true"/>
          <a:Position>
            <a:X>0</a:X>
            <a:Y>0</a:Y>
          </a:Position>
          <a:Z>0</a:Z>
        </a:ViewElement>
      </a:ViewElems>
    </a:Result>
  </GetEditSlideResult>
</GetEditSlideResponse>
</s:Body>
</s:Envelope>

```

Next, the protocol client sends a request to the protocol server to modify the text of a shape on the presentation slide that was referenced in the previous step. The identifier of the shape, 2, is specified by the **ShapeId** element of the first element in the **EditShapes** element of the **EditSlide** element (section [2.2.4.29](#)). The following **IPptEdit_ReplaceText_InputMessage** message is sent to the protocol server:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <ReplaceText
xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServer
InternalService/">
      <presentationId>d=F39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a9a007e0723241fm568ced57
b5cc4aeda74fff96066a8267m</presentationId>
      <slideId>256</slideId>
      <shapeId>2</shapeId>
      <newText>&lt;p class=nonListPara style='direction:ltr' lang=en-US style='language:en-
US' align=center&gt;&lt;font size=6 face=Calibri color="#7293C9"&gt;&lt;span style='font-
variant:normal;text-transform:uppercase'&gt;&lt;b&gt;Fancy Title with
3d.&lt;/b&gt;&lt;/span&gt;&lt;/font&gt;&lt;/p&gt;</newText>
    </ReplaceText>
  </s:Body>
</s:Envelope>

```

The protocol server responds with the result of the operation. The following **IPptEdit_ReplaceText_OutputMessage** message is sent to the protocol client:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <ReplaceTextResponse
xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServer
InternalService/">
      <ReplaceTextResult 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="a:EditCommandResponse">
          <a:PresChanges>
            <a:AddedSlideInfos/>
            <a:AddedSlides/>
            <a:MainMasterIds i:nil="true"/>
            <a:ModifiedSlides>
              <a:EditSlideChanges>

```



```

    <a:DelElems
xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
    <a:Elem>
      <a:ViewElement>
        <a:Alt/>
        <a:Clip>
          <a:Bottom>205</a:Bottom>
          <a:Left>0</a:Left>
          <a:Right>407</a:Right>
          <a:Top>0</a:Top>
        </a:Clip>
        <a:Html i:nil="true"/>
        <a:Id>2</a:Id>
    </a:Elem>
    <a:ImgId>./editImageHandler.ashx?Pid=d%3DF39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a
9a007e0723241fm568ced57b5cc4aeda74fff96066a8267m&amp;Rid=iv0q3epq%2Edro%2Epng</a:ImgId>
      <a:Position>
        <a:X>149</a:X>
        <a:Y>8</a:Y>
      </a:Position>
      <a:Z>1</a:Z>
    </a:ViewElement>
    <a:ViewElement>
      <a:Alt i:nil="true"/>
      <a:Clip>
        <a:Bottom>102</a:Bottom>
        <a:Left>407</a:Left>
        <a:Right>659</a:Right>
        <a:Top>0</a:Top>
      </a:Clip>
      <a:Html i:nil="true"/>
      <a:Id>3</a:Id>
    </a:ViewElement>
    <a:ImgId>./editImageHandler.ashx?Pid=d%3DF39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a
9a007e0723241fm568ced57b5cc4aeda74fff96066a8267m&amp;Rid=iv0q3epq%2Edro%2Epng</a:ImgId>
      <a:Position>
        <a:X>199</a:X>
        <a:Y>55</a:Y>
      </a:Position>
      <a:Z>2</a:Z>
    </a:ViewElement>
  </a:Elem>
  <a:Hidden>>false</a:Hidden>
  <a:Id>256</a:Id>
  <a:ModifiedShapes>
    <a>EditShape>
      <a:Alt i:nil="true"/>
      <a>EditBounds>
        <a:Bottom>140</a:Bottom>
        <a:Left>46</a:Left>
        <a:Right>573</a:Right>
        <a:Top>41</a:Top>
      </a>EditBounds>
      <a>EditHtml>&lt;div class=defaultPHStyles style='font-size:32.0px;font-
family:Calibri;color:#7293C9;text-align:center'&gt; &lt;/div&gt;&lt;p class=nonListPara
style='direction:ltr' lang=en-US style='language:en-US' align=center&gt;&lt;font size=6
face=Calibri color="#7293C9"&gt;&lt;span style='font-variant:normal;text-
transform:uppercase'&gt;&lt;b&gt;Fancy Title with
3d.&lt;/b&gt;&lt;/span&gt;&lt;/font&gt;&lt;/p&gt;</a>EditHtml>
      <a:FontSizeMapping>
        <a:Size1>12</a:Size1>
        <a:Size2>15</a:Size2>
        <a:Size3>19</a:Size3>
        <a:Size4>21</a:Size4>
        <a:Size5>28</a:Size5>
        <a:Size6>44</a:Size6>
        <a:Size7>56</a:Size7>
      </a:FontSizeMapping>
      <a:IsPicture>>false</a:IsPicture>
    </a>EditShape>
  </a:ModifiedShapes>

```

```

        <a:IsPlaceholder>>true</a:IsPlaceholder>
        <a:IsPrompt>>false</a:IsPrompt>
        <a:IsSmartArt>>false</a:IsSmartArt>
        <a:PlaceholderType>2</a:PlaceholderType>
        <a:ShapeId>2</a:ShapeId>
        <a:ShapeViewElemId>2</a:ShapeViewElemId>
        <a:SmartArtInfo i:nil="true"/>
        <a:TextViewElemId>3</a:TextViewElemId>
    </a>EditShape>
</a:ModifiedShapes>
<a:Notes i:nil="true"/>
<a:Refresh>>false</a:Refresh>
<a:Thumb>
    <a:Alt i:nil="true"/>
    <a:Clip>
        <a:Bottom>96</a:Bottom>
        <a:Left>659</a:Left>
        <a:Right>787</a:Right>
        <a:Top>0</a:Top>
    </a:Clip>
    <a:Html i:nil="true"/>
    <a:Id>5</a:Id>
<a:ImgId>./editImageHandler.ashx?Pid=d%3DF39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a9a007e0723241fm568ced57b5cc4aeda74fff96066a8267m&amp;Rid=iv0q3epq%2Edro%2Epng</a:ImgId>
    <a:Position>
        <a:X>0</a:X>
        <a:Y>0</a:Y>
    </a:Position>
    <a:Z>0</a:Z>
    </a:Thumb>
    <a>Title>Fancy Title with 3d.</a>Title>
</a>EditSlideChanges>
</a:ModifiedSlides>
<a:Redo>>false</a:Redo>
<a:SlideOrder i:nil="true"/>
<a:Undo>>true</a:Undo>
</a:PresChanges>
</a:Result>
</ReplaceTextResult>
</ReplaceTextResponse>
</s:Body>
</s:Envelope>

```

Finally, the protocol client sends a request to the protocol server to save the changes made to the presentation and end the session. The following **IPptEdit_SaveAndClose_InputMessage** message is sent to the protocol server:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <SaveAndClose
      xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/">
      <presentationId>d=F39bc64b0ef8e43c493bc0191f7fb5371m371f1619a41e40dd8a9a007e0723241fmcf8e360c9a82447c9a2ce13a5987d49cm</presentationId>
      <fReleaseLock>true</fReleaseLock>
      <fWaitForSave>true</fWaitForSave>
    </SaveAndClose>
  </s:Body>
</s:Envelope>

```

The protocol server responds with the result of the operation. The following **IPptEdit_SaveAndClose_OutputMessage** message is sent to the protocol client:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <SaveAndCloseResponse
xmlns="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServer
InternalService/">
      <SaveAndCloseResult 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="a:EditCommandResponse">
          <a:PresChanges i:nil="true"/>
        </a:Result>
      </SaveAndCloseResult>
    </SaveAndCloseResponse>
  </s:Body>
</s:Envelope>
```

5 Security

5.1 Security Considerations for Implementers

The **Html** child element of **ViewElement** (section [2.2.4.51](#)), the **Html** child element of **Notes** (section [2.2.4.37](#)), and the **EditHtml** child element of **EditShape** (section [2.2.4.28](#)) contain [\[HTML\]](#) fragments. HTML fragments are susceptible to **cross-site scripting** attacks. Implementers are advised to sanitize these HTML fragments when generating or processing these elements.

Some operations which specify a change to a presentation such as **DeleteSlide** (section [3.1.4.17](#)) are susceptible to **cross-site request forgery** attacks. Implementers are advised to use a canary [<101>](#) to mitigate operations from this type of attack.

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

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

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

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions
  xmlns:tns4="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
  erverInternalService/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
  targetNamespace="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPoint
  EditServerInternalService/" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <xs:schema
      xmlns:tns5="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
      erverInternalService/Imports"
      targetNamespace="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPoint
      EditServerInternalService/Imports">
      <xs:import
        namespace="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interfa
        ce.Shared"/>
      <xs:import
        namespace="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.In
        terface"/>
      <xs:import namespace="http://schemas.datacontract.org/2004/07/p"/>
      <xs:import namespace="http://schemas.microsoft.com/2003/10/Serialization/" />
      <xs:import namespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays" />
      <xs:import
        namespace="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditSe
        rverInternalService/" />
      </xs:schema>
    </wsdl:types>
    <wsdl:portType name="IPptEdit">
      <wsdl:operation name="LogULS">
        <wsdl:input
          wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
          ServerInternalService/IPptEdit/LogULS" message="tns4:IPptEdit_LogULS_InputMessage"/>
        <wsdl:output
          wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
          ServerInternalService/IPptEdit/LogULSResponse" message="tns4:IPptEdit_LogULS_OutputMessage"/>
        </wsdl:operation>
      <wsdl:operation name="GetPresentationId">
        <wsdl:input
          wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
          ServerInternalService/IPptEdit/GetPresentationId"
          message="tns4:IPptEdit_GetPresentationId_InputMessage"/>
        <wsdl:output
          wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
          ServerInternalService/IPptEdit/GetPresentationIdResponse"
          message="tns4:IPptEdit_GetPresentationId_OutputMessage"/>
        </wsdl:operation>
      <wsdl:operation name="GetCoauthUpdates">
        <wsdl:input
          wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
          ServerInternalService/IPptEdit/GetCoauthUpdates"
          message="tns4:IPptEdit_GetCoauthUpdates_InputMessage"/>
        <wsdl:output
          wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
          ServerInternalService/IPptEdit/GetCoauthUpdatesResponse"
          message="tns4:IPptEdit_GetCoauthUpdates_OutputMessage"/>
        </wsdl:operation>
      <wsdl:operation name="GetEditPresInfo">
        <wsdl:input
          wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
          ServerInternalService/IPptEdit/GetEditPresInfo"
          message="tns4:IPptEdit_GetEditPresInfo_InputMessage"/>
```

```

    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/GetEditPresInfoResponse"
message="tns4:IPptEdit_GetEditPresInfo_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="GetEditPresInfoGetEditSlideById">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/GetEditPresInfoGetEditSlideById"
message="tns4:IPptEdit_GetEditPresInfoGetEditSlideById_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/GetEditPresInfoGetEditSlideByIdResponse"
message="tns4:IPptEdit_GetEditPresInfoGetEditSlideById_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="GetEditPresResources">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/GetEditPresResources"
message="tns4:IPptEdit_GetEditPresResources_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/GetEditPresResourcesResponse"
message="tns4:IPptEdit_GetEditPresResources_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="GetEditSlide">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/GetEditSlide"
message="tns4:IPptEdit_GetEditSlide_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/GetEditSlideResponse"
message="tns4:IPptEdit_GetEditSlide_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="MoveSlide">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/MoveSlide" message="tns4:IPptEdit_MoveSlide_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/MoveSlideResponse"
message="tns4:IPptEdit_MoveSlide_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="InsertSlide">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/InsertSlide"
message="tns4:IPptEdit_InsertSlide_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/InsertSlideResponse"
message="tns4:IPptEdit_InsertSlide_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ReplaceText">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReplaceText"
message="tns4:IPptEdit_ReplaceText_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReplaceTextResponse"
message="tns4:IPptEdit_ReplaceText_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetAnimation">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetAnimation"
message="tns4:IPptEdit_SetAnimation_InputMessage"/>

```

```

    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetAnimationResponse"
message="tns4:IPptEdit_SetAnimation_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetTransition">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetTransition"
message="tns4:IPptEdit_SetTransition_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetTransitionResponse"
message="tns4:IPptEdit_SetTransition_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetTransitionApplyAll">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetTransitionApplyAll"
message="tns4:IPptEdit_SetTransitionApplyAll_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetTransitionApplyAllResponse"
message="tns4:IPptEdit_SetTransitionApplyAll_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ReorderAnimation">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReorderAnimation"
message="tns4:IPptEdit_ReorderAnimation_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReorderAnimationResponse"
message="tns4:IPptEdit_ReorderAnimation_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ClearPlaceholder">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ClearPlaceholder"
message="tns4:IPptEdit_ClearPlaceholder_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ClearPlaceholderResponse"
message="tns4:IPptEdit_ClearPlaceholder_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="DeleteSlide">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/DeleteSlide"
message="tns4:IPptEdit_DeleteSlide_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/DeleteSlideResponse"
message="tns4:IPptEdit_DeleteSlide_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="DuplicateSlide">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/DuplicateSlide"
message="tns4:IPptEdit_DuplicateSlide_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/DuplicateSlideResponse"
message="tns4:IPptEdit_DuplicateSlide_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ShowHideSlide">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit

```

```

ServerInternalService/IPptEdit/ShowHideSlide"
message="tns4:IPptEdit ShowHideSlide InputMessage"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ShowHideSlideResponse"
message="tns4:IPptEdit_ShowHideSlide_OutputMessage"/>
  </wsdl:operation>
  <wsdl:operation name="ReplaceNotes">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReplaceNotes"
message="tns4:IPptEdit_ReplaceNotes_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReplaceNotesResponse"
message="tns4:IPptEdit ReplaceNotes OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="Undo">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/Undo" message="tns4:IPptEdit_Undo_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/UndoResponse" message="tns4:IPptEdit_Undo_OutputMessage"/>
      </wsdl:operation>
      <wsdl:operation name="Redo">
        <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/Redo" message="tns4:IPptEdit_Redo_InputMessage"/>
        <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/RedoResponse" message="tns4:IPptEdit_Redo_OutputMessage"/>
        </wsdl:operation>
        <wsdl:operation name="ChangePictureStyle">
          <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ChangePictureStyle"
message="tns4:IPptEdit_ChangePictureStyle_InputMessage"/>
          <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ChangePictureStyleResponse"
message="tns4:IPptEdit_ChangePictureStyle_OutputMessage"/>
          </wsdl:operation>
          <wsdl:operation name="ResetPicture">
            <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ResetPicture"
message="tns4:IPptEdit_ResetPicture_InputMessage"/>
            <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ResetPictureResponse"
message="tns4:IPptEdit ResetPicture OutputMessage"/>
            </wsdl:operation>
            <wsdl:operation name="MoveShape">
              <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/MoveShape" message="tns4:IPptEdit_MoveShape_InputMessage"/>
              <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/MoveShapeResponse"
message="tns4:IPptEdit_MoveShape_OutputMessage"/>
              </wsdl:operation>
              <wsdl:operation name="ResizeShape">
                <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ResizeShape"
message="tns4:IPptEdit_ResizeShape_InputMessage"/>
                <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit

```



```

ServerInternalService/IPptEdit/ResizeShapeResponse"
message="tns4:IPptEdit ResizeShape OutputMessage"/>
  </wsdl:operation>
  <wsdl:operation name="RotateShape">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/RotateShape"
message="tns4:IPptEdit RotateShape InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/RotateShapeResponse"
message="tns4:IPptEdit_RotateShape_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="FlipShape">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/FlipShape" message="tns4:IPptEdit_FlipShape_InputMessage"/>
        <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/FlipShapeResponse"
message="tns4:IPptEdit_FlipShape_OutputMessage"/>
      </wsdl:operation>
      <wsdl:operation name="InsertShape">
        <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/InsertShape"
message="tns4:IPptEdit InsertShape InputMessage"/>
          <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/InsertShapeResponse"
message="tns4:IPptEdit_InsertShape_OutputMessage"/>
        </wsdl:operation>
        <wsdl:operation name="DuplicateShape">
          <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/DuplicateShape"
message="tns4:IPptEdit_DuplicateShape_InputMessage"/>
            <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/DuplicateShapeResponse"
message="tns4:IPptEdit_DuplicateShape_OutputMessage"/>
          </wsdl:operation>
          <wsdl:operation name="ArrangeShape">
            <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ArrangeShape"
message="tns4:IPptEdit_ArrangeShape_InputMessage"/>
              <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ArrangeShapeResponse"
message="tns4:IPptEdit_ArrangeShape_OutputMessage"/>
            </wsdl:operation>
            <wsdl:operation name="UngroupShape">
              <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/UngroupShape"
message="tns4:IPptEdit_UngroupShape_InputMessage"/>
                <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/UngroupShapeResponse"
message="tns4:IPptEdit_UngroupShape_OutputMessage"/>
              </wsdl:operation>
              <wsdl:operation name="ApplyShapeStyle">
                <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeStyle"
message="tns4:IPptEdit_ApplyShapeStyle_InputMessage"/>
                  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit

```

```

ServerInternalService/IPptEdit/ApplyShapeStyleResponse"
message="tns4:IPptEdit ApplyShapeStyle OutputMessage"/>
  </wsdl:operation>
  <wsdl:operation name="ApplyShapeFill">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeFill"
message="tns4:IPptEdit ApplyShapeFill InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeFillResponse"
message="tns4:IPptEdit_ApplyShapeFill_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="RemoveShapeFill">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/RemoveShapeFill"
message="tns4:IPptEdit_RemoveShapeFill_InputMessage"/>
        <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/RemoveShapeFillResponse"
message="tns4:IPptEdit_RemoveShapeFill_OutputMessage"/>
      </wsdl:operation>
      <wsdl:operation name="ApplyShapeOutlineColor">
        <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeOutlineColor"
message="tns4:IPptEdit_ApplyShapeOutlineColor_InputMessage"/>
          <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeOutlineColorResponse"
message="tns4:IPptEdit_ApplyShapeOutlineColor_OutputMessage"/>
        </wsdl:operation>
        <wsdl:operation name="ApplyShapeOutlineWidth">
          <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeOutlineWidth"
message="tns4:IPptEdit_ApplyShapeOutlineWidth_InputMessage"/>
            <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeOutlineWidthResponse"
message="tns4:IPptEdit_ApplyShapeOutlineWidth_OutputMessage"/>
          </wsdl:operation>
          <wsdl:operation name="ApplyShapeOutlineDashStyle">
            <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeOutlineDashStyle"
message="tns4:IPptEdit_ApplyShapeOutlineDashStyle_InputMessage"/>
              <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeOutlineDashStyleResponse"
message="tns4:IPptEdit ApplyShapeOutlineDashStyle OutputMessage"/>
            </wsdl:operation>
            <wsdl:operation name="ApplyShapeOutlineEndStyle">
              <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeOutlineEndStyle"
message="tns4:IPptEdit ApplyShapeOutlineEndStyle InputMessage"/>
                <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyShapeOutlineEndStyleResponse"
message="tns4:IPptEdit ApplyShapeOutlineEndStyle OutputMessage"/>
              </wsdl:operation>
              <wsdl:operation name="RemoveShapeOutline">
                <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/RemoveShapeOutline"
message="tns4:IPptEdit RemoveShapeOutline InputMessage"/>
                  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit

```

```

ServerInternalService/IPptEdit/RemoveShapeOutlineResponse"
message="tns4:IPptEdit RemoveShapeOutline OutputMessage"/>
  </wsdl:operation>
  <wsdl:operation name="SaveAndClose">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SaveAndClose"
message="tns4:IPptEdit SaveAndClose InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SaveAndCloseResponse"
message="tns4:IPptEdit_SaveAndClose_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="UpdateCanary">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/UpdateCanary"
message="tns4:IPptEdit_UpdateCanary_InputMessage"/>
        <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/UpdateCanaryResponse"
message="tns4:IPptEdit_UpdateCanary_OutputMessage"/>
      </wsdl:operation>
      <wsdl:operation name="Print">
        <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/Print" message="tns4:IPptEdit Print InputMessage"/>
          <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/PrintResponse" message="tns4:IPptEdit_Print_OutputMessage"/>
        </wsdl:operation>
        <wsdl:operation name="ChangeSmartArtLayout">
          <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ChangeSmartArtLayout"
message="tns4:IPptEdit_ChangeSmartArtLayout_InputMessage"/>
            <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ChangeSmartArtLayoutResponse"
message="tns4:IPptEdit_ChangeSmartArtLayout_OutputMessage"/>
          </wsdl:operation>
          <wsdl:operation name="ChangeSmartArtColor">
            <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ChangeSmartArtColor"
message="tns4:IPptEdit_ChangeSmartArtColor_InputMessage"/>
              <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ChangeSmartArtColorResponse"
message="tns4:IPptEdit_ChangeSmartArtColor_OutputMessage"/>
            </wsdl:operation>
            <wsdl:operation name="ChangeSmartArtStyle">
              <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ChangeSmartArtStyle"
message="tns4:IPptEdit_ChangeSmartArtStyle InputMessage"/>
                <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ChangeSmartArtStyleResponse"
message="tns4:IPptEdit_ChangeSmartArtStyle_OutputMessage"/>
              </wsdl:operation>
              <wsdl:operation name="InsertSmartArt">
                <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/InsertSmartArt"
message="tns4:IPptEdit_InsertSmartArt_InputMessage"/>
                  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/InsertSmartArtResponse"
message="tns4:IPptEdit_InsertSmartArt_OutputMessage"/>
                </wsdl:operation>

```

```

    </wsdl:operation>
    <wsdl:operation name="ResetSmartArt">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ResetSmartArt"
message="tns4:IPptEdit_ResetSmartArt_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ResetSmartArtResponse"
message="tns4:IPptEdit_ResetSmartArt_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ReverseSmartArt">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReverseSmartArt"
message="tns4:IPptEdit_ReverseSmartArt_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ReverseSmartArtResponse"
message="tns4:IPptEdit_ReverseSmartArt_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ApplyTheme">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyTheme" message="tns4:IPptEdit_ApplyTheme_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ApplyThemeResponse"
message="tns4:IPptEdit_ApplyTheme_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetWordArt">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetWordArt" message="tns4:IPptEdit_SetWordArt_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetWordArtResponse"
message="tns4:IPptEdit_SetWordArt_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="InsertClipart">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/InsertClipart"
message="tns4:IPptEdit_InsertClipart_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/InsertClipartResponse"
message="tns4:IPptEdit_InsertClipart_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="PastePicture">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/PastePicture"
message="tns4:IPptEdit_PastePicture_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/PastePictureResponse"
message="tns4:IPptEdit_PastePicture_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ShapeFormatPainting">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ShapeFormatPainting"
message="tns4:IPptEdit_ShapeFormatPainting_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ShapeFormatPaintingResponse"
message="tns4:IPptEdit_ShapeFormatPainting_OutputMessage"/>
    </wsdl:operation>

```

```

    <wsdl:operation name="AddComment">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/AddComment" message="tns4:IPptEdit_AddComment_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/AddCommentResponse"
message="tns4:IPptEdit_AddComment_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="DeleteComment">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/DeleteComment"
message="tns4:IPptEdit_DeleteComment_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/DeleteCommentResponse"
message="tns4:IPptEdit_DeleteComment_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="EditComment">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/EditComment"
message="tns4:IPptEdit_EditComment_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/EditCommentResponse"
message="tns4:IPptEdit_EditComment_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="MoveComment">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/MoveComment"
message="tns4:IPptEdit_MoveComment_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/MoveCommentResponse"
message="tns4:IPptEdit_MoveComment_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetShapeBold">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeBold"
message="tns4:IPptEdit_SetShapeBold_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeBoldResponse"
message="tns4:IPptEdit_SetShapeBold_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetShapeItalic">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeItalic"
message="tns4:IPptEdit_SetShapeItalic_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeItalicResponse"
message="tns4:IPptEdit_SetShapeItalic_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetShapeUnderline">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeUnderline"
message="tns4:IPptEdit_SetShapeUnderline_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeUnderlineResponse"
message="tns4:IPptEdit_SetShapeUnderline_OutputMessage"/>
    </wsdl:operation>

```

```

    <wsdl:operation name="SetShapeAlignment">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeAlignment"
message="tns4:IPptEdit_SetShapeAlignment_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeAlignmentResponse"
message="tns4:IPptEdit_SetShapeAlignment_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetShapeFontName">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeFontName"
message="tns4:IPptEdit_SetShapeFontName_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeFontNameResponse"
message="tns4:IPptEdit_SetShapeFontName_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetShapeFontSize">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeFontSize"
message="tns4:IPptEdit_SetShapeFontSize_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeFontSizeResponse"
message="tns4:IPptEdit_SetShapeFontSize_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetShapeFontColor">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeFontColor"
message="tns4:IPptEdit_SetShapeFontColor_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeFontColorResponse"
message="tns4:IPptEdit_SetShapeFontColor_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetShapeBullet">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeBullet"
message="tns4:IPptEdit_SetShapeBullet_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeBulletResponse"
message="tns4:IPptEdit_SetShapeBullet_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetShapeTextDirection">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeTextDirection"
message="tns4:IPptEdit_SetShapeTextDirection_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/SetShapeTextDirectionResponse"
message="tns4:IPptEdit_SetShapeTextDirection_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ChangeLayout">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ChangeLayout"
message="tns4:IPptEdit_ChangeLayout_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEdit
ServerInternalService/IPptEdit/ChangeLayoutResponse"
message="tns4:IPptEdit_ChangeLayout_OutputMessage"/>

```

```

    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="DefaultBinding_IPptEdit" type="tns4:IPptEdit">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="LogULS">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/LogULS" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetPresentationId">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/GetPresentationId" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetCoauthUpdates">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/GetCoauthUpdates" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetEditPresInfo">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/GetEditPresInfo" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetEditPresInfoGetEditSlideById">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/GetEditPresInfoGetEditSlideById" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetEditPresResources">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/GetEditPresResources" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
</wsdl:service>

```

```

</wsdl:operation>
<wsdl:operation name="GetEditSlide">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/GetEditSlide" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="MoveSlide">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/MoveSlide" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="InsertSlide">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/InsertSlide" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ReplaceText">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ReplaceText" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetAnimation">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetAnimation" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetTransition">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetTransition" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetTransitionApplyAll">

```



```

    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetTransitionApplyAll" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ReorderAnimation">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ReorderAnimation" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ClearPlaceholder">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ClearPlaceholder" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DeleteSlide">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/DeleteSlide" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DuplicateSlide">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/DuplicateSlide" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ShowHideSlide">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ShowHideSlide" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ReplaceNotes">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ReplaceNotes" style="document"/>

```

```

    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="Undo">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/Undo" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
<wsdl:operation name="Redo">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/Redo" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
<wsdl:operation name="ChangePictureStyle">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ChangePictureStyle" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
<wsdl:operation name="ResetPicture">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ResetPicture" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
<wsdl:operation name="MoveShape">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/MoveShape" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
<wsdl:operation name="ResizeShape">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ResizeShape" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>

```

```

        <wsdl:output>
          <soap:body use="literal"/>
        </wsdl:output>
      </wsdl:operation>
    <wsdl:operation name="RotateShape">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/RotateShape" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="FlipShape">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/FlipShape" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="InsertShape">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/InsertShape" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="DuplicateShape">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/DuplicateShape" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="ArrangeShape">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ArrangeShape" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="UngroupShape">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/UngroupShape" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:output>

```

```

</wsdl:operation>
<wsdl:operation name="ApplyShapeStyle">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ApplyShapeStyle" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ApplyShapeFill">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ApplyShapeFill" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="RemoveShapeFill">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/RemoveShapeFill" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ApplyShapeOutlineColor">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ApplyShapeOutlineColor" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ApplyShapeOutlineWidth">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ApplyShapeOutlineWidth" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ApplyShapeOutlineDashStyle">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ApplyShapeOutlineDashStyle" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ApplyShapeOutlineEndStyle">

```

```

    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ApplyShapeOutlineEndStyle" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="RemoveShapeOutline">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/RemoveShapeOutline" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SaveAndClose">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SaveAndClose" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="UpdateCanary">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/UpdateCanary" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="Print">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/Print" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ChangeSmartArtLayout">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ChangeSmartArtLayout" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ChangeSmartArtColor">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ChangeSmartArtColor" style="document"/>

```

```

    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ChangeSmartArtStyle">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ChangeSmartArtStyle" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="InsertSmartArt">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/InsertSmartArt" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ResetSmartArt">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ResetSmartArt" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ReverseSmartArt">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ReverseSmartArt" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ApplyTheme">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ApplyTheme" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetWordArt">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetWordArt" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>

```

```

        <wsdl:output>
          <soap:body use="literal"/>
        </wsdl:output>
      </wsdl:operation>
    <wsdl:operation name="InsertClipart">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/InsertClipart" style="document"/>
        <wsdl:input>
          <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal"/>
        </wsdl:output>
      </wsdl:operation>
    <wsdl:operation name="PastePicture">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/PastePicture" style="document"/>
        <wsdl:input>
          <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal"/>
        </wsdl:output>
      </wsdl:operation>
    <wsdl:operation name="ShapeFormatPainting">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ShapeFormatPainting" style="document"/>
        <wsdl:input>
          <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal"/>
        </wsdl:output>
      </wsdl:operation>
    <wsdl:operation name="AddComment">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/AddComment" style="document"/>
        <wsdl:input>
          <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal"/>
        </wsdl:output>
      </wsdl:operation>
    <wsdl:operation name="DeleteComment">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/DeleteComment" style="document"/>
        <wsdl:input>
          <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal"/>
        </wsdl:output>
      </wsdl:operation>
    <wsdl:operation name="EditComment">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/EditComment" style="document"/>
        <wsdl:input>
          <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal"/>
        </wsdl:output>
      </wsdl:operation>

```

```

</wsdl:operation>
<wsdl:operation name="MoveComment">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/MoveComment" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetShapeBold">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetShapeBold" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetShapeItalic">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetShapeItalic" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetShapeUnderline">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetShapeUnderline" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetShapeAlignment">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetShapeAlignment" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetShapeFontName">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetShapeFontName" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetShapeFontSize">

```



```

    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetShapeFontSize" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetShapeFontColor">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetShapeFontColor" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetShapeBullet">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetShapeBullet" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetShapeTextDirection">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/SetShapeTextDirection" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ChangeLayout">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditS
erverInternalService/IPptEdit/ChangeLayout" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:message name="IPptEdit_AddComment_InputMessage">
    <wsdl:part name="parameters" element="tns4:AddComment"/>
</wsdl:message>
<wsdl:message name="IPptEdit AddComment OutputMessage">
    <wsdl:part name="parameters" element="tns4:AddCommentResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ApplyShapeFill_InputMessage">
    <wsdl:part name="parameters" element="tns4:ApplyShapeFill"/>
</wsdl:message>
<wsdl:message name="IPptEdit ApplyShapeFill OutputMessage">
    <wsdl:part name="parameters" element="tns4:ApplyShapeFillResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ApplyShapeOutlineColor_InputMessage">
    <wsdl:part name="parameters" element="tns4:ApplyShapeOutlineColor"/>

```

```

</wsdl:message>
<wsdl:message name="IPptEdit_ApplyShapeOutlineColor_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ApplyShapeOutlineColorResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ApplyShapeOutlineDashStyle_InputMessage">
  <wsdl:part name="parameters" element="tns4:ApplyShapeOutlineDashStyle"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ApplyShapeOutlineDashStyle_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ApplyShapeOutlineDashStyleResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ApplyShapeOutlineEndStyle_InputMessage">
  <wsdl:part name="parameters" element="tns4:ApplyShapeOutlineEndStyle"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ApplyShapeOutlineEndStyle_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ApplyShapeOutlineEndStyleResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ApplyShapeOutlineWidth_InputMessage">
  <wsdl:part name="parameters" element="tns4:ApplyShapeOutlineWidth"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ApplyShapeOutlineWidth_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ApplyShapeOutlineWidthResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ApplyShapeStyle_InputMessage">
  <wsdl:part name="parameters" element="tns4:ApplyShapeStyle"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ApplyShapeStyle_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ApplyShapeStyleResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ApplyTheme_InputMessage">
  <wsdl:part name="parameters" element="tns4:ApplyTheme"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ApplyTheme_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ApplyThemeResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ArrangeShape_InputMessage">
  <wsdl:part name="parameters" element="tns4:ArrangeShape"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ArrangeShape_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ArrangeShapeResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ChangeLayout_InputMessage">
  <wsdl:part name="parameters" element="tns4:ChangeLayout"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ChangeLayout_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ChangeLayoutResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ChangePictureStyle_InputMessage">
  <wsdl:part name="parameters" element="tns4:ChangePictureStyle"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ChangePictureStyle_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ChangePictureStyleResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ChangeSmartArtColor_InputMessage">
  <wsdl:part name="parameters" element="tns4:ChangeSmartArtColor"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ChangeSmartArtColor_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ChangeSmartArtColorResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ChangeSmartArtLayout_InputMessage">
  <wsdl:part name="parameters" element="tns4:ChangeSmartArtLayout"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ChangeSmartArtLayout_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ChangeSmartArtLayoutResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ChangeSmartArtStyle_InputMessage">
  <wsdl:part name="parameters" element="tns4:ChangeSmartArtStyle"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ChangeSmartArtStyle_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ChangeSmartArtStyleResponse"/>

```

```

</wsdl:message>
<wsdl:message name="IPptEdit_ClearPlaceholder_InputMessage">
  <wsdl:part name="parameters" element="tns4:ClearPlaceholder"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ClearPlaceholder_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ClearPlaceholderResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_DeleteComment_InputMessage">
  <wsdl:part name="parameters" element="tns4>DeleteComment"/>
</wsdl:message>
<wsdl:message name="IPptEdit_DeleteComment_OutputMessage">
  <wsdl:part name="parameters" element="tns4>DeleteCommentResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_DeleteSlide_InputMessage">
  <wsdl:part name="parameters" element="tns4>DeleteSlide"/>
</wsdl:message>
<wsdl:message name="IPptEdit_DeleteSlide_OutputMessage">
  <wsdl:part name="parameters" element="tns4>DeleteSlideResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_DuplicateShape_InputMessage">
  <wsdl:part name="parameters" element="tns4:DuplicateShape"/>
</wsdl:message>
<wsdl:message name="IPptEdit_DuplicateShape_OutputMessage">
  <wsdl:part name="parameters" element="tns4:DuplicateShapeResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_DuplicateSlide_InputMessage">
  <wsdl:part name="parameters" element="tns4:DuplicateSlide"/>
</wsdl:message>
<wsdl:message name="IPptEdit_DuplicateSlide_OutputMessage">
  <wsdl:part name="parameters" element="tns4:DuplicateSlideResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_EditComment_InputMessage">
  <wsdl:part name="parameters" element="tns4>EditComment"/>
</wsdl:message>
<wsdl:message name="IPptEdit_EditComment_OutputMessage">
  <wsdl:part name="parameters" element="tns4>EditCommentResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_FlipShape_InputMessage">
  <wsdl:part name="parameters" element="tns4:FlipShape"/>
</wsdl:message>
<wsdl:message name="IPptEdit_FlipShape_OutputMessage">
  <wsdl:part name="parameters" element="tns4:FlipShapeResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_GetCoauthUpdates_InputMessage">
  <wsdl:part name="parameters" element="tns4:GetCoauthUpdates"/>
</wsdl:message>
<wsdl:message name="IPptEdit_GetCoauthUpdates_OutputMessage">
  <wsdl:part name="parameters" element="tns4:GetCoauthUpdatesResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_GetEditPresInfo_InputMessage">
  <wsdl:part name="parameters" element="tns4:GetEditPresInfo"/>
</wsdl:message>
<wsdl:message name="IPptEdit_GetEditPresInfo_OutputMessage">
  <wsdl:part name="parameters" element="tns4:GetEditPresInfoResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_GetEditPresInfoGetEditSlideById_InputMessage">
  <wsdl:part name="parameters" element="tns4:GetEditPresInfoGetEditSlideById"/>
</wsdl:message>
<wsdl:message name="IPptEdit_GetEditPresInfoGetEditSlideById_OutputMessage">
  <wsdl:part name="parameters" element="tns4:GetEditPresInfoGetEditSlideByIdResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_GetEditPresResources_InputMessage">
  <wsdl:part name="parameters" element="tns4:GetEditPresResources"/>
</wsdl:message>
<wsdl:message name="IPptEdit_GetEditPresResources_OutputMessage">
  <wsdl:part name="parameters" element="tns4:GetEditPresResourcesResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_GetEditSlide_InputMessage">
  <wsdl:part name="parameters" element="tns4:GetEditSlide"/>

```

```

</wsdl:message>
<wsdl:message name="IPptEdit_GetEditSlide_OutputMessage">
  <wsdl:part name="parameters" element="tns4:GetEditSlideResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_GetPresentationId_InputMessage">
  <wsdl:part name="parameters" element="tns4:GetPresentationId"/>
</wsdl:message>
<wsdl:message name="IPptEdit_GetPresentationId_OutputMessage">
  <wsdl:part name="parameters" element="tns4:GetPresentationIdResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_InsertClipart_InputMessage">
  <wsdl:part name="parameters" element="tns4:InsertClipart"/>
</wsdl:message>
<wsdl:message name="IPptEdit_InsertClipart_OutputMessage">
  <wsdl:part name="parameters" element="tns4:InsertClipartResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_InsertShape_InputMessage">
  <wsdl:part name="parameters" element="tns4:InsertShape"/>
</wsdl:message>
<wsdl:message name="IPptEdit_InsertShape_OutputMessage">
  <wsdl:part name="parameters" element="tns4:InsertShapeResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_InsertSlide_InputMessage">
  <wsdl:part name="parameters" element="tns4:InsertSlide"/>
</wsdl:message>
<wsdl:message name="IPptEdit_InsertSlide_OutputMessage">
  <wsdl:part name="parameters" element="tns4:InsertSlideResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_InsertSmartArt_InputMessage">
  <wsdl:part name="parameters" element="tns4:InsertSmartArt"/>
</wsdl:message>
<wsdl:message name="IPptEdit_InsertSmartArt_OutputMessage">
  <wsdl:part name="parameters" element="tns4:InsertSmartArtResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_LogULS_InputMessage">
  <wsdl:part name="parameters" element="tns4:LogULS"/>
</wsdl:message>
<wsdl:message name="IPptEdit_LogULS_OutputMessage">
  <wsdl:part name="parameters" element="tns4:LogULSResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_MoveComment_InputMessage">
  <wsdl:part name="parameters" element="tns4:MoveComment"/>
</wsdl:message>
<wsdl:message name="IPptEdit_MoveComment_OutputMessage">
  <wsdl:part name="parameters" element="tns4:MoveCommentResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_MoveShape_InputMessage">
  <wsdl:part name="parameters" element="tns4:MoveShape"/>
</wsdl:message>
<wsdl:message name="IPptEdit_MoveShape_OutputMessage">
  <wsdl:part name="parameters" element="tns4:MoveShapeResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_MoveSlide_InputMessage">
  <wsdl:part name="parameters" element="tns4:MoveSlide"/>
</wsdl:message>
<wsdl:message name="IPptEdit_MoveSlide_OutputMessage">
  <wsdl:part name="parameters" element="tns4:MoveSlideResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_PastePicture_InputMessage">
  <wsdl:part name="parameters" element="tns4:PastePicture"/>
</wsdl:message>
<wsdl:message name="IPptEdit_PastePicture_OutputMessage">
  <wsdl:part name="parameters" element="tns4:PastePictureResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_Print_InputMessage">
  <wsdl:part name="parameters" element="tns4:Print"/>
</wsdl:message>
<wsdl:message name="IPptEdit_Print_OutputMessage">
  <wsdl:part name="parameters" element="tns4:PrintResponse"/>

```

```

</wsdl:message>
<wsdl:message name="IPptEdit_Redo_InputMessage">
  <wsdl:part name="parameters" element="tns4:Redo"/>
</wsdl:message>
<wsdl:message name="IPptEdit_Redo_OutputMessage">
  <wsdl:part name="parameters" element="tns4:RedoResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_RemoveShapeFill_InputMessage">
  <wsdl:part name="parameters" element="tns4:RemoveShapeFill"/>
</wsdl:message>
<wsdl:message name="IPptEdit_RemoveShapeFill_OutputMessage">
  <wsdl:part name="parameters" element="tns4:RemoveShapeFillResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_RemoveShapeOutline_InputMessage">
  <wsdl:part name="parameters" element="tns4:RemoveShapeOutline"/>
</wsdl:message>
<wsdl:message name="IPptEdit_RemoveShapeOutline_OutputMessage">
  <wsdl:part name="parameters" element="tns4:RemoveShapeOutlineResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ReorderAnimation_InputMessage">
  <wsdl:part name="parameters" element="tns4:ReorderAnimation"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ReorderAnimation_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ReorderAnimationResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ReplaceNotes_InputMessage">
  <wsdl:part name="parameters" element="tns4:ReplaceNotes"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ReplaceNotes_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ReplaceNotesResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ReplaceText_InputMessage">
  <wsdl:part name="parameters" element="tns4:ReplaceText"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ReplaceText_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ReplaceTextResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ResetPicture_InputMessage">
  <wsdl:part name="parameters" element="tns4:ResetPicture"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ResetPicture_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ResetPictureResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ResetSmartArt_InputMessage">
  <wsdl:part name="parameters" element="tns4:ResetSmartArt"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ResetSmartArt_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ResetSmartArtResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ResizeShape_InputMessage">
  <wsdl:part name="parameters" element="tns4:ResizeShape"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ResizeShape_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ResizeShapeResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ReverseSmartArt_InputMessage">
  <wsdl:part name="parameters" element="tns4:ReverseSmartArt"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ReverseSmartArt_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ReverseSmartArtResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_RotateShape_InputMessage">
  <wsdl:part name="parameters" element="tns4:RotateShape"/>
</wsdl:message>
<wsdl:message name="IPptEdit_RotateShape_OutputMessage">
  <wsdl:part name="parameters" element="tns4:RotateShapeResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SaveAndClose_InputMessage">
  <wsdl:part name="parameters" element="tns4:SaveAndClose"/>

```

```

</wsdl:message>
<wsdl:message name="IPptEdit_SaveAndClose_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SaveAndCloseResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetAnimation_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetAnimation"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetAnimation_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetAnimationResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeAlignment_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeAlignment"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeAlignment_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeAlignmentResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeBold_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeBold"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeBold_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeBoldResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeBullet_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeBullet"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeBullet_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeBulletResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeFontColor_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeFontColor"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeFontColor_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeFontColorResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeFontName_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeFontName"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeFontName_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeFontNameResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeFontSize_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeFontSize"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeFontSize_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeFontSizeResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeItalic_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeItalic"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeItalic_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeItalicResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeTextDirection_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeTextDirection"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeTextDirection_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeTextDirectionResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeUnderline_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeUnderline"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetShapeUnderline_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetShapeUnderlineResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetTransition_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetTransition"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetTransition_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetTransitionResponse"/>

```

```

</wsdl:message>
<wsdl:message name="IPptEdit_SetTransitionApplyAll_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetTransitionApplyAll"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetTransitionApplyAll_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetTransitionApplyAllResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetWordArt_InputMessage">
  <wsdl:part name="parameters" element="tns4:SetWordArt"/>
</wsdl:message>
<wsdl:message name="IPptEdit_SetWordArt_OutputMessage">
  <wsdl:part name="parameters" element="tns4:SetWordArtResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ShapeFormatPainting_InputMessage">
  <wsdl:part name="parameters" element="tns4:ShapeFormatPainting"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ShapeFormatPainting_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ShapeFormatPaintingResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ShowHideSlide_InputMessage">
  <wsdl:part name="parameters" element="tns4:ShowHideSlide"/>
</wsdl:message>
<wsdl:message name="IPptEdit_ShowHideSlide_OutputMessage">
  <wsdl:part name="parameters" element="tns4:ShowHideSlideResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_Undo_InputMessage">
  <wsdl:part name="parameters" element="tns4:Undo"/>
</wsdl:message>
<wsdl:message name="IPptEdit_Undo_OutputMessage">
  <wsdl:part name="parameters" element="tns4:UndoResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_UngroupShape_InputMessage">
  <wsdl:part name="parameters" element="tns4:UngroupShape"/>
</wsdl:message>
<wsdl:message name="IPptEdit_UngroupShape_OutputMessage">
  <wsdl:part name="parameters" element="tns4:UngroupShapeResponse"/>
</wsdl:message>
<wsdl:message name="IPptEdit_UpdateCanary_InputMessage">
  <wsdl:part name="parameters" element="tns4:UpdateCanary"/>
</wsdl:message>
<wsdl:message name="IPptEdit_UpdateCanary_OutputMessage">
  <wsdl:part name="parameters" element="tns4:UpdateCanaryResponse"/>
</wsdl:message>
</wsdl:definitions>

```

7 Appendix B: Full XML Schema

Schema name	Prefix	Section
http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared	tns	7.1
http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.Interface	tns1	7.2
http://schemas.datacontract.org/2004/07/p	tns2	7.3
http://schemas.microsoft.com/2003/10/Serialization/Arrays	q3	7.4
http://schemas.microsoft.com/2003/10/Serialization/	tns3	7.5
http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/	tns4	7.6

For ease of implementation, the following sections provide the full XML schema for this protocol.

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

```
<?xml version="1.0" encoding="UTF-8"?>
<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:simpleType name="ErrorCode">
    <xs:restriction base="xs:string">
      <xs:enumeration value="ErrorRESERVEDSuccess"/>
      <xs:enumeration value="ErrorRESERVEDUnknown"/>
      <xs:enumeration value="ErrorCannotOpenDocumentFileType"/>
      <xs:enumeration value="ErrorCannotOpenDocumentIRM"/>
      <xs:enumeration value="ErrorCannotOpenDocumentPassword"/>
      <xs:enumeration value="ErrorCannotOpenDocument"/>
      <xs:enumeration value="ErrorCannotInstallFonts"/>
      <xs:enumeration value="ErrorCannotConvertModifyPassword"/>
      <xs:enumeration value="ErrorCannotComposeSlideParse"/>
      <xs:enumeration value="ErrorCannotComposeSlideUnexpected"/>
      <xs:enumeration value="ErrorWacConversionFailure"/>
      <xs:enumeration value="ErrorResourceNotGenerated"/>
      <xs:enumeration value="ErrorGenericSlideCorruption"/>
      <xs:enumeration value="ErrorFileNotFound"/>
      <xs:enumeration value="ErrorGeneralUnknown"/>
      <xs:enumeration value="ErrorFailedToOpenDocument"/>
      <xs:enumeration value="ErrorFailedToRetrieveResource"/>
      <xs:enumeration value="ErrorFailedToRenderDocument"/>
      <xs:enumeration value="ErrorFailedToRenderSlide"/>
      <xs:enumeration value="ErrorFailedToSaveDocument"/>
      <xs:enumeration value="ErrorCannotSaveDocument"/>
      <xs:enumeration value="ErrorInvalidCommand"/>
      <xs:enumeration value="ErrorDocVersionMismatch"/>
      <xs:enumeration value="ErrorCannotAcquirePptEditLock"/>
      <xs:enumeration value="ErrorRequiresCheckout"/>
      <xs:enumeration value="ErrorLockedForOfflineCheckout"/>
      <xs:enumeration value="ErrorLockedByAnotherUser"/>
      <xs:enumeration value="ErrorAccessDenied"/>
      <xs:enumeration value="ErrorNoLicense"/>
      <xs:enumeration value="ErrorRequestTooBig"/>
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
```



```

<xs:enumeration value="ErrorTextTooBig"/>
<xs:enumeration value="ErrorTryAgain"/>
<xs:enumeration value="ErrorUnexpected"/>
<xs:enumeration value="ErrorCannotOpenDocumentMacro"/>
<xs:enumeration value="ErrorCannotOpenDocumentFinal"/>
<xs:enumeration value="ErrorCannotOpenDocumentSigned"/>
<xs:enumeration value="ErrorCannotOpenEmbeddedFonts"/>
<xs:enumeration value="ErrorNoAvailableEditAppServer"/>
<xs:enumeration value="ErrorServiceUnavailable"/>
<xs:enumeration value="ErrorPptEditExistingExclusiveLock"/>
<xs:enumeration value="ErrorPptEditCheckedOutByAnother"/>
<xs:enumeration value="ErrorCannotOpenDocumentGatekeeperFailure"/>
<xs:enumeration value="ErrorEditWorkerTimeout"/>
<xs:enumeration value="ErrorEditSlideCorruption"/>
<xs:enumeration value="ErrorServiceBusy"/>
<xs:enumeration value="ErrorClipartCannotConnect"/>
<xs:enumeration value="ErrorClipartCannotDownload"/>
<xs:enumeration value="ErrorMediaNotEnabled"/>
<xs:enumeration value="ErrorUnsupportedMediaFormat"/>
<xs:enumeration value="ErrorLinkedMediaNotEnabled"/>
<xs:enumeration value="ErrorUnsupportedLinkTarget"/>
<xs:enumeration value="ErrorEmbeddedMediaNotEnabled"/>
<xs:enumeration value="ErrorEmbeddedMediaExceedsMaxSize"/>
<xs:enumeration value="ErrorTranscodingNotEnabled"/>
<xs:enumeration value="ErrorMediaUnavailable"/>
<xs:enumeration value="ErrorCannotOpenODPDocumentPassword"/>
<xs:enumeration value="ErrorLocalChangeLostDueToCoauthorEdit"/>
<xs:enumeration value="ErrorNumberOfCoauthorsReachedMax"/>
<xs:enumeration value="ErrorCannotAcquirePptEditSharedLock"/>
<xs:enumeration value="ErrorInProgress"/>
<xs:enumeration value="ErrorImageTooLarge"/>
<xs:enumeration value="ErrorImageWrongFormat"/>
<xs:enumeration value="ErrorImageNoAccess"/>
<xs:enumeration value="ErrorWacItemRetrievalDocumentOutputNotFound"/>
<xs:enumeration value="ErrorWacItemRetrievalAccessDenied"/>
<xs:enumeration value="ErrorWacItemRetrievalCommunicationErrorStreamDirty"/>
<xs:enumeration value="ErrorWacItemRetrievalCommunicationErrorStreamClean"/>
<xs:enumeration value="ErrorWacItemRetrievalUnknownError"/>
<xs:enumeration value="ErrorWacConversionManagerDoesNotSupportThisRequest"/>
<xs:enumeration value="ErrorWacConversionCannotFindSourceDocument"/>
<xs:enumeration value="ErrorWacConversionWorkerException"/>
<xs:enumeration value="ErrorWacConversionWorkerCrashed"/>
<xs:enumeration value="ErrorWacConversionWorkerHung"/>
<xs:enumeration value="ErrorWacConversionUnknownResponse"/>
<xs:enumeration value="ErrorWacConversionUnknownError"/>
<xs:enumeration value="ErrorWacConversionFirstServerResult"/>
<xs:enumeration value="ErrorCannotOpenDocumentWrongExt"/>
<xs:enumeration value="ErrorCannotOpenDocumentStgDocFileCorrupt"/>
<xs:enumeration value="ErrorCannotOpenDocumentStgFileAlreadyExists"/>
<xs:enumeration value="ErrorCannotOpenDocumentMmcfPackagePartNotFound"/>
<xs:enumeration value="ErrorCannotOpenDocumentArchiveError"/>
<xs:enumeration value="ErrorCannotPastePictureCannotCreateUrl"/>
<xs:enumeration value="ErrorCannotPastePictureInvalidResponse"/>
<xs:enumeration value="ErrorCannotPastePictureNotFound"/>
<xs:enumeration value="ErrorCannotPastePictureResponseEmpty"/>
<xs:enumeration value="ErrorCannotPastePictureUnexpectedStatusCode"/>
<xs:enumeration value="ErrorCannotPastePictureUnexpectedWebException"/>
<xs:enumeration value="ErrorRESERVEDLastError"/>
</xs:restriction>
</xs:simpleType>
<xs:element name="ErrorCode" nillable="true" type="tns:ErrorCode"/>
</xs:schema>

```

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

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<xs:schema
xmlns:tns1="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.I
nterface" 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:import namespace="http://schemas.microsoft.com/2003/10/Serialization/" />
  <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>
  <xs:element name="ClientActions" nillable="true" type="tns1:ClientActions" />
</xs:schema>

```

7.3 http://schemas.datacontract.org/2004/07/p Schema

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:tns2="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="ServiceResult">
    <xs:sequence>
      <xs:element minOccurs="0" name="Error" nillable="true" type="tns2:ServiceError" />
      <xs:element minOccurs="0" name="Result" nillable="true" type="xs:anyType" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ServiceResult" nillable="true" type="tns2:ServiceResult" />
  <xs:complexType name="ServiceError">
    <xs:sequence>
      <xs:element
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interfa
ce.Shared" minOccurs="0" name="Code" type="tns:ErrorCode" />
      <xs:element minOccurs="0" name="Message" nillable="true" type="xs:string" />
      <xs:element
xmlns:tns1="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.I
nterface" minOccurs="0" name="RecommendedActions" type="tns1:ClientActions" />
      <xs:element minOccurs="0" name="Title" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="Type" type="tns2:ServiceErrorType" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ServiceError" nillable="true" type="tns2: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:enumeration value="SessionFull" />
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="ServiceErrorType" nillable="true" type="tns2:ServiceErrorType" />
  <xs:complexType name="CoauthUpdate">
    <xs:sequence>
      <xs:element minOccurs="0" name="State" nillable="true" type="tns2:CoauthState" />
      <xs:element minOccurs="0" name="Token" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="CoauthUpdate" nillable="true" type="tns2:CoauthUpdate" />
  <xs:complexType name="CoauthState">

```

```

    <xs:sequence>
      <xs:element minOccurs="0" name="cs" type="xs:int"/>
      <xs:element minOccurs="0" name="et" nillable="true" type="tns2:ArrayOfEditorInfo"/>
      <xs:element minOccurs="0" name="lsave" type="xs:dateTime"/>
      <xs:element minOccurs="0" name="lsmce" type="xs:dateTime"/>
      <xs:element minOccurs="0" name="lsum" type="xs:dateTime"/>
      <xs:element minOccurs="0" name="meid" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="sui" nillable="true" type="tns2:ContentUpdateInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="CoauthState" nillable="true" type="tns2:CoauthState"/>
  <xs:complexType name="ArrayOfEditorInfo">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="EditorInfo" nillable="true"
type="tns2:EditorInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfEditorInfo" nillable="true" type="tns2:ArrayOfEditorInfo"/>
  <xs:complexType name="EditorInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="EmailAddress" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="ID" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="SIPAddress" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="TimeOut" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="UserName" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="EditorInfo" nillable="true" type="tns2:EditorInfo"/>
  <xs:complexType name="ContentUpdateInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="leid" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="lplu" type="xs:dateTime"/>
      <xs:element minOccurs="0" name="redo" type="xs:boolean"/>
      <xs:element minOccurs="0" name="sui" nillable="true"
type="tns2:ArrayOfSlideUpdateInfo"/>
      <xs:element minOccurs="0" name="undo" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ContentUpdateInfo" nillable="true" type="tns2:ContentUpdateInfo"/>
  <xs:complexType name="ArrayOfSlideUpdateInfo">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="SlideUpdateInfo" nillable="true"
type="tns2:SlideUpdateInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfSlideUpdateInfo" nillable="true"
type="tns2:ArrayOfSlideUpdateInfo"/>
  <xs:complexType name="SlideUpdateInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="leid" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="lsu" type="xs:dateTime"/>
      <xs:element minOccurs="0" name="sid" type="tns2:SlideID"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="SlideUpdateInfo" nillable="true" type="tns2:SlideUpdateInfo"/>
  <xs:complexType name="SlideID">
    <xs:sequence>
      <xs:element minOccurs="0" name="Cid" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="Sid" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="SlideID" nillable="true" type="tns2:SlideID"/>
  <xs:complexType name="EditPresentationInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="CompatProblems" type="xs:boolean"/>
      <xs:element minOccurs="0" name="EditSlideInfos" nillable="true"
type="tns2:ArrayOfEditSlideInfoFe"/>
      <xs:element minOccurs="0" name="FirstSlideIndex" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>

```

```

    <xs:element minOccurs="0" name="MainMasters" nillable="true"
type="tns2:ArrayOfMainMasterFe"/>
    <xs:element minOccurs="0" name="Redo" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Rtl" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Size" nillable="true" type="tns2:Point"/>
    <xs:element minOccurs="0" name="Slide" nillable="true" type="tns2:EditSlide"/>
    <xs:element minOccurs="0" name="SlideOrder" nillable="true" type="tns2:SlideIdList"/>
    <xs:element minOccurs="0" name="Undo" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="EditPresentationInfo" nillable="true" type="tns2:EditPresentationInfo"/>
<xs:complexType name="ArrayOfEditSlideInfoFe">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="EditSlideInfoFe" nillable="true"
type="tns2:EditSlideInfoFe"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ArrayOfEditSlideInfoFe" nillable="true"
type="tns2:ArrayOfEditSlideInfoFe"/>
<xs:complexType name="EditSlideInfoFe">
  <xs:sequence>
    <xs:element minOccurs="0" name="HasAnimationOrTransition" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Hidden" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Id" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Thumb" nillable="true" type="tns2:ViewElement"/>
    <xs:element minOccurs="0" name="Title" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="EditSlideInfoFe" nillable="true" type="tns2:EditSlideInfoFe"/>
<xs:complexType name="ViewElement">
  <xs:sequence>
    <xs:element minOccurs="0" name="Alt" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Clip" nillable="true" type="tns2:ClippingInfo"/>
    <xs:element minOccurs="0" name="Html" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Id" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="ImgId" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Position" nillable="true" type="tns2:Point"/>
    <xs:element minOccurs="0" name="Z" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ViewElement" nillable="true" type="tns2:ViewElement"/>
<xs:complexType name="ClippingInfo">
  <xs:sequence>
    <xs:element minOccurs="0" name="Bounds" nillable="true" type="tns2:Rectangle"/>
    <xs:element minOccurs="0" name="TotalHeight" type="xs:int"/>
    <xs:element minOccurs="0" name="TotalWidth" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ClippingInfo" nillable="true" type="tns2:ClippingInfo"/>
<xs:complexType name="Rectangle">
  <xs:sequence>
    <xs:element minOccurs="0" name="Bottom" type="xs:int"/>
    <xs:element minOccurs="0" name="Left" type="xs:int"/>
    <xs:element minOccurs="0" name="Right" type="xs:int"/>
    <xs:element minOccurs="0" name="Top" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="Rectangle" nillable="true" type="tns2:Rectangle"/>
<xs:complexType name="Point">
  <xs:sequence>
    <xs:element minOccurs="0" name="X" type="xs:int"/>
    <xs:element minOccurs="0" name="Y" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="Point" nillable="true" type="tns2:Point"/>
<xs:complexType name="ArrayOfMainMasterFe">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="MainMasterFe" nillable="true"
type="tns2:MainMasterFe"/>
  </xs:sequence>
</xs:complexType>

```

```

    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfMainMasterFe" nillable="true" type="tns2:ArrayOfMainMasterFe"/>
  <xs:complexType name="MainMasterFe">
    <xs:sequence>
      <xs:element minOccurs="0" name="ContentMasters" nillable="true"
type="tns2:ArrayOfContentMasterFe"/>
      <xs:element minOccurs="0" name="Id" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="InContent" type="xs:boolean"/>
      <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="ThemeFamilyId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="VariantIndex" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="MainMasterFe" nillable="true" type="tns2:MainMasterFe"/>
  <xs:complexType name="ArrayOfContentMasterFe">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="ContentMasterFe" nillable="true"
type="tns2:ContentMasterFe"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfContentMasterFe" nillable="true"
type="tns2:ArrayOfContentMasterFe"/>
  <xs:complexType name="ContentMasterFe">
    <xs:sequence>
      <xs:element minOccurs="0" name="Id" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="SlideLayoutType" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ContentMasterFe" nillable="true" type="tns2:ContentMasterFe"/>
  <xs:complexType name="EditSlide">
    <xs:sequence>
      <xs:element minOccurs="0" name="Comments" nillable="true" type="tns2:ArrayOfComment"/>
      <xs:element minOccurs="0" name="ContentMasterId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="Created" type="xs:dateTime"/>
      <xs:element minOccurs="0" name="EditAnims" nillable="true"
type="tns2:ArrayOfEditAnimInfo"/>
      <xs:element minOccurs="0" name="EditShapes" nillable="true"
type="tns2:ArrayOfEditShape"/>
      <xs:element minOccurs="0" name="EditTransition" nillable="true"
type="tns2:EditTransitionInfo"/>
      <xs:element minOccurs="0" name="Hidden" type="xs:boolean"/>
      <xs:element minOccurs="0" name="ID" type="tns2:SlideID"/>
      <xs:element minOccurs="0" name="IsTitleContentMaster" type="xs:boolean"/>
      <xs:element minOccurs="0" name="MainMasterId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="Notes" nillable="true" type="tns2:Notes"/>
      <xs:element minOccurs="0" name="SlideId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="Theme" nillable="true" type="tns2:Theme"/>
      <xs:element minOccurs="0" name="ThumbnailViewElement" nillable="true"
type="tns2:ViewElement"/>
      <xs:element minOccurs="0" name="ViewElems" nillable="true"
type="tns2:ArrayOfViewElement"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="EditSlide" nillable="true" type="tns2:EditSlide"/>
  <xs:complexType name="ArrayOfComment">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="Comment" nillable="true"
type="tns2:Comment"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfComment" nillable="true" type="tns2:ArrayOfComment"/>
  <xs:complexType name="Comment">
    <xs:sequence>
      <xs:element minOccurs="0" name="Children" nillable="true" type="tns2:ArrayOfComment"/>
      <xs:element minOccurs="0" name="authorId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="authorName" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="dt" type="xs:dateTime"/>
    </xs:sequence>
  </xs:complexType>

```

```

        <xs:element minOccurs="0" name="idx" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="pos" nillable="true" type="tns2:Point"/>
        <xs:element minOccurs="0" name="text" nillable="true" type="xs:string"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="Comment" nillable="true" type="tns2:Comment"/>
<xs:complexType name="ArrayOfEditAnimInfo">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="EditAnimInfo" nillable="true"
type="tns2:EditAnimInfo"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="ArrayOfEditAnimInfo" nillable="true" type="tns2:ArrayOfEditAnimInfo"/>
<xs:complexType name="EditAnimInfo">
    <xs:sequence>
        <xs:element xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
minOccurs="0" name="BuildIds" nillable="true" type="q3:ArrayOfunsignedInt"/>
        <xs:element minOccurs="0" name="Direction" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="EffectId" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="HasExisting" type="xs:boolean"/>
        <xs:element minOccurs="0" name="HasTrigger" type="xs:boolean"/>
        <xs:element minOccurs="0" name="SequenceId" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="ShapeId" type="xs:unsignedInt"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="EditAnimInfo" nillable="true" type="tns2:EditAnimInfo"/>
<xs:complexType name="ArrayOfEditShape">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="EditShape" nillable="true"
type="tns2:EditShape"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="ArrayOfEditShape" nillable="true" type="tns2:ArrayOfEditShape"/>
<xs:complexType name="EditShape">
    <xs:sequence>
        <xs:element minOccurs="0" name="Alt" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="EditBounds" nillable="true" type="tns2:Rectangle"/>
        <xs:element minOccurs="0" name="EditHtml" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="FFlippedHorizontally" type="xs:boolean"/>
        <xs:element minOccurs="0" name="FFlippedVertically" type="xs:boolean"/>
        <xs:element minOccurs="0" name="FontName" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="FontSize" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="FontSizeMapping" nillable="true"
type="tns2:FontSizeMapping"/>
        <xs:element minOccurs="0" name="HasEffectAndNoFill" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsActiveX" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsBold" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsChart" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsConnected" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsGroup" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsInk" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsItalic" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsLine" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsNonNativeVideo" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsOleObject" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsPicture" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsPlaceholder" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsPrompt" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsSmartArt" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsStraightLine" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsTable" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsTextbox" type="xs:boolean"/>
        <xs:element minOccurs="0" name="IsUnderline" type="xs:boolean"/>
        <xs:element minOccurs="0" name="PlaceholderType" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="RotatedBoundingRegion" nillable="true"
type="tns2:BoundingRegion"/>
        <xs:element minOccurs="0" name="Rotation" type="xs:int"/>
        <xs:element minOccurs="0" name="ShapeId" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="ShapeViewElemId" type="xs:unsignedInt"/>
    </xs:sequence>
</xs:complexType>

```

```

    <xs:element minOccurs="0" name="SmartArtInfo" nillable="true"
type="tns2:SmartArtInfo"/>
    <xs:element minOccurs="0" name="TextAlignmentType" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="TextBulletType" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="TextDirectionType" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="TextViewElemId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="X" type="xs:double"/>
    <xs:element minOccurs="0" name="Y" type="xs:double"/>
    <xs:element minOccurs="0" name="Z" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="EditShape" nillable="true" type="tns2:EditShape"/>
<xs:complexType name="FontSizeMapping">
  <xs:sequence>
    <xs:element minOccurs="0" name="Size1" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Size2" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Size3" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Size4" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Size5" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Size6" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Size7" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="FontSizeMapping" nillable="true" type="tns2:FontSizeMapping"/>
<xs:complexType name="BoundingRegion">
  <xs:sequence>
    <xs:element minOccurs="0" name="BottomLeft" nillable="true" type="tns2:Point"/>
    <xs:element minOccurs="0" name="BottomRight" nillable="true" type="tns2:Point"/>
    <xs:element minOccurs="0" name="TopLeft" nillable="true" type="tns2:Point"/>
    <xs:element minOccurs="0" name="TopRight" nillable="true" type="tns2:Point"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="BoundingRegion" nillable="true" type="tns2:BoundingRegion"/>
<xs:complexType name="SmartArtInfo">
  <xs:sequence>
    <xs:element minOccurs="0" name="fReverseChecked" type="xs:boolean"/>
    <xs:element minOccurs="0" name="fReverseEnabled" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="SmartArtInfo" nillable="true" type="tns2:SmartArtInfo"/>
<xs:complexType name="EditTransitionInfo">
  <xs:sequence>
    <xs:element minOccurs="0" name="Direction" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="Duration" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="IsManualAdvance" type="xs:boolean"/>
    <xs:element minOccurs="0" name="TransitionType" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="EditTransitionInfo" nillable="true" type="tns2:EditTransitionInfo"/>
<xs:complexType name="Notes">
  <xs:sequence>
    <xs:element minOccurs="0" name="Html" nillable="true" type="tns2:Html"/>
    <xs:element minOccurs="0" name="IsPrompt" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="Notes" nillable="true" type="tns2:Notes"/>
<xs:complexType name="Html">
  <xs:sequence>
    <xs:element minOccurs="0" name="Text" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="Html" nillable="true" type="tns2:Html"/>
<xs:complexType name="Theme">
  <xs:sequence>
    <xs:element minOccurs="0" name="Accent1" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent2" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent3" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent4" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent5" nillable="true" type="tns2:ThemeColor"/>
  </xs:sequence>

```

```

    <xs:element minOccurs="0" name="Accent6" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Dark1" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Dark2" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Light1" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Light2" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="PresetSchemeColorMapping" nillable="true"
type="tns2:PresetSchemeColorMapping"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="Theme" nillable="true" type="tns2:Theme"/>
<xs:complexType name="ThemeColor">
  <xs:sequence>
    <xs:element minOccurs="0" name="b" type="xs:unsignedByte"/>
    <xs:element minOccurs="0" name="g" type="xs:unsignedByte"/>
    <xs:element minOccurs="0" name="r" type="xs:unsignedByte"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ThemeColor" nillable="true" type="tns2:ThemeColor"/>
<xs:complexType name="PresetSchemeColorMapping">
  <xs:sequence>
    <xs:element minOccurs="0" name="Accent1" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent2" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent3" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent4" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent5" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Accent6" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Background1" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Background2" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Text1" nillable="true" type="tns2:ThemeColor"/>
    <xs:element minOccurs="0" name="Text2" nillable="true" type="tns2:ThemeColor"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="PresetSchemeColorMapping" nillable="true"
type="tns2:PresetSchemeColorMapping"/>
<xs:complexType name="ArrayOfViewElement">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="ViewElement" nillable="true"
type="tns2:ViewElement"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ArrayOfViewElement" nillable="true" type="tns2:ArrayOfViewElement"/>
<xs:complexType name="SlideIdList">
  <xs:sequence>
    <xs:element xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
minOccurs="0" name="SlideIds" nillable="true" type="q3:ArrayOfunsignedInt"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="SlideIdList" nillable="true" type="tns2:SlideIdList"/>
<xs:complexType name="EditPresentationResources">
  <xs:sequence>
    <xs:element minOccurs="0" name="MainMasterResources" nillable="true"
type="tns2:ArrayOfMainMasterResourcesFe"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="EditPresentationResources" nillable="true"
type="tns2:EditPresentationResources"/>
<xs:complexType name="ArrayOfMainMasterResourcesFe">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="MainMasterResourcesFe"
nillable="true" type="tns2:MainMasterResourcesFe"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ArrayOfMainMasterResourcesFe" nillable="true"
type="tns2:ArrayOfMainMasterResourcesFe"/>
<xs:complexType name="MainMasterResourcesFe">
  <xs:sequence>
    <xs:element minOccurs="0" name="LineStyles" nillable="true"
type="tns2:ArrayOfStyleGalleryItem"/>
    <xs:element minOccurs="0" name="MasterId" type="xs:unsignedInt"/>
  </xs:sequence>

```



```

        <xs:element minOccurs="0" name="ShapeStyles" nillable="true"
type="tns2:ArrayOfStyleGalleryItem"/>
    </xs:sequence>
</xs:complexType>
    <xs:element name="MainMasterResourcesFe" nillable="true"
type="tns2:MainMasterResourcesFe"/>
    <xs:complexType name="ArrayOfStyleGalleryItem">
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" name="StyleGalleryItem" nillable="true"
type="tns2:StyleGalleryItem"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="ArrayOfStyleGalleryItem" nillable="true"
type="tns2:ArrayOfStyleGalleryItem"/>
    <xs:complexType name="StyleGalleryItem">
        <xs:sequence>
            <xs:element minOccurs="0" name="Icon" nillable="true" type="tns2:ViewElement"/>
            <xs:element minOccurs="0" name="StyleId" type="xs:unsignedInt"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="StyleGalleryItem" nillable="true" type="tns2:StyleGalleryItem"/>
    <xs:complexType name="EditCommandResponse">
        <xs:sequence>
            <xs:element minOccurs="0" name="PresChanges" nillable="true"
type="tns2:EditPresentationChanges"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="EditCommandResponse" nillable="true" type="tns2:EditCommandResponse"/>
    <xs:complexType name="EditPresentationChanges">
        <xs:sequence>
            <xs:element minOccurs="0" name="AddedSlideInfos" nillable="true"
type="tns2:ArrayOfEditSlideInfoFe"/>
            <xs:element minOccurs="0" name="AddedSlides" nillable="true"
type="tns2:ArrayOfEditSlide"/>
            <xs:element minOccurs="0" name="InvalidateAllThumbs" type="xs:boolean"/>
            <xs:element minOccurs="0" name="MainMasterIds" nillable="true"
type="tns2:SlideIdList"/>
            <xs:element minOccurs="0" name="ModifiedSlides" nillable="true"
type="tns2:ArrayOfEditSlideChanges"/>
            <xs:element minOccurs="0" name="Redo" type="xs:boolean"/>
            <xs:element minOccurs="0" name="SlideOrder" nillable="true" type="tns2:SlideIdList"/>
            <xs:element minOccurs="0" name="Undo" type="xs:boolean"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="EditPresentationChanges" nillable="true"
type="tns2:EditPresentationChanges"/>
    <xs:complexType name="ArrayOfEditSlide">
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" name="EditSlide" nillable="true"
type="tns2:EditSlide"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="ArrayOfEditSlide" nillable="true" type="tns2:ArrayOfEditSlide"/>
    <xs:complexType name="ArrayOfEditSlideChanges">
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" name="EditSlideChanges" nillable="true"
type="tns2:EditSlideChanges"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="ArrayOfEditSlideChanges" nillable="true"
type="tns2:ArrayOfEditSlideChanges"/>
    <xs:complexType name="EditSlideChanges">
        <xs:sequence>
            <xs:element minOccurs="0" name="Comments" nillable="true" type="tns2:ArrayOfComment"/>
            <xs:element xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
minOccurs="0" name="DelElems" nillable="true" type="q3:ArrayOfunsignedInt"/>
            <xs:element minOccurs="0" name="EditAnims" nillable="true"
type="tns2:ArrayOfEditAnimInfo"/>

```

```

    <xs:element minOccurs="0" name="EditTransition" nillable="true"
type="tns2:EditTransitionInfo"/>
    <xs:element minOccurs="0" name="Elems" nillable="true" type="tns2:ArrayOfViewElement"/>
    <xs:element minOccurs="0" name="HasAnimationOrTransition" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Hidden" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Id" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="ModifiedShapes" nillable="true"
type="tns2:ArrayOfEditShape"/>
    <xs:element minOccurs="0" name="Notes" nillable="true" type="tns2:Notes"/>
    <xs:element minOccurs="0" name="Refresh" type="xs:boolean"/>
    <xs:element minOccurs="0" name="Thumb" nillable="true" type="tns2:ViewElement"/>
    <xs:element minOccurs="0" name="Title" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="EditSlideChanges" nillable="true" type="tns2:EditSlideChanges"/>
<xs:complexType name="PptViewingService.PrintResult">
  <xs:sequence>
    <xs:element minOccurs="0" name="PrintUrl" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="PptViewingService.PrintResult" nillable="true"
type="tns2:PptViewingService.PrintResult"/>
<xs:import
namespace="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared"/>
<xs:import
namespace="http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.Interface"/>
<xs:import namespace="http://schemas.microsoft.com/2003/10/Serialization/" />
<xs:import namespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
</xs:schema>

```

7.4 http://schemas.microsoft.com/2003/10/Serialization/Arrays Schema

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:q3="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="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="q3:ArrayOfunsignedInt"/>
</xs:schema>

```

7.5 http://schemas.microsoft.com/2003/10/Serialization/ Schema

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:tns3="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="tns3:char"/>
<xs:simpleType name="char">
  <xs:restriction base="xs:int"/>
</xs:simpleType>
<xs:element name="duration" nillable="true" type="tns3: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="tns3: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>

```

7.6 <http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/> Schema

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPoint
EditServerInternalService/" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://schemas.datacontract.org/2004/07/p"/>
  <xs:element name="LogULS">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="tagID" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="level" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="message" nillable="true" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="LogULSResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="LogULSResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetPresentationId">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="rawDocId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="backendUriOverride" nillable="true"
type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetPresentationIdResponse">

```

```

    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="GetPresentationIdResult" nillable="true"
type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetCoauthUpdates">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="lastToken" nillable="true" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetCoauthUpdatesResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetCoauthUpdatesResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetEditPresInfo">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="firstSlideIndex" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="numSlides" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="includedEditSlideIndex" type="xs:int"/>
        <xs:element minOccurs="0" name="clientWidth" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="clientHeight" type="xs:unsignedInt"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetEditPresInfoResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetEditPresInfoResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetEditPresInfoGetEditSlideById">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="firstSlideIndex" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="numSlides" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="includedEditSlideId" nillable="true"
type="xs:string"/>
        <xs:element minOccurs="0" name="clientWidth" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="clientHeight" type="xs:unsignedInt"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetEditPresInfoGetEditSlideByIdResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetEditPresInfoGetEditSlideByIdResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetEditPresResources">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

    </xs:complexType>
  </xs:element>
  <xs:element name="GetEditPresResourcesResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetEditPresResourcesResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetEditSlide">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetEditSlideResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="GetEditSlideResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="MoveSlide">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="srcId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="dstId" nillable="true" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="MoveSlideResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="MoveSlideResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="InsertSlide">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="dstId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="mainMasterId" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="contentMasterId" type="xs:unsignedInt"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="InsertSlideResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="InsertSlideResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ReplaceText">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="newText" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="partialEdit" type="xs:boolean"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ReplaceTextResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ReplaceTextResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetAnimation">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="fxId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="direction" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="sequenceId" type="xs:unsignedInt"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetAnimationResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetAnimationResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetTransition">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="type" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="direction" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="duration" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="isManualAdvance" type="xs:boolean"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetTransitionResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetTransitionResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetTransitionApplyAll">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetTransitionApplyAllResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetTransitionApplyAllResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ReorderAnimation">
    <xs:complexType>

```

```

    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="fLater" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ReorderAnimationResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ReorderAnimationResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ClearPlaceholder">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ClearPlaceholderResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ClearPlaceholderResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="DeleteSlide">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="DeleteSlideResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="DeleteSlideResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="DuplicateSlide">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="srcId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="dstId" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="DuplicateSlideResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="DuplicateSlideResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ShowHideSlide">
  <xs:complexType>
    <xs:sequence>

```

```

        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="fHidden" type="xs:boolean"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ShowHideSlideResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ShowHideSlideResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ReplaceNotes">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="newNotes" nillable="true" type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ReplaceNotesResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ReplaceNotesResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="Undo">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="UndoResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="UndoResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="Redo">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="RedoResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="RedoResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ChangePictureStyle">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="pictureStyleId" type="xs:unsignedInt"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>

```



```

    </xs:complexType>
  </xs:element>
  <xs:element name="ChangePictureStyleResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ChangePictureStyleResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ResetPicture">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ResetPictureResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ResetPictureResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="MoveShape">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="x" type="xs:double"/>
        <xs:element minOccurs="0" name="y" type="xs:double"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="MoveShapeResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="MoveShapeResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ResizeShape">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="deltaNorth" type="xs:double"/>
        <xs:element minOccurs="0" name="deltaSouth" type="xs:double"/>
        <xs:element minOccurs="0" name="deltaWest" type="xs:double"/>
        <xs:element minOccurs="0" name="deltaEast" type="xs:double"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ResizeShapeResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ResizeShapeResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="RotateShape">
    <xs:complexType>

```

```

    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="degrees" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="RotateShapeResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="RotateShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="FlipShape">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="fHorizontal" type="xs:boolean"/>
      <xs:element minOccurs="0" name="fVertical" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="FlipShapeResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="FlipShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="InsertShape">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeType" type="xs:int"/>
      <xs:element minOccurs="0" name="shapeCtrPosX" type="xs:int"/>
      <xs:element minOccurs="0" name="shapeCtrPosY" type="xs:int"/>
      <xs:element minOccurs="0" name="shapeWidth" type="xs:double"/>
      <xs:element minOccurs="0" name="shapeHeight" type="xs:double"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="InsertShapeResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="InsertShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="DuplicateShape">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="sourceSlideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="sourceShapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="destinationSlideId" nillable="true"
type="xs:string"/>
      <xs:element minOccurs="0" name="cloneShapePosX" type="xs:int"/>
      <xs:element minOccurs="0" name="cloneShapePosY" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

<xs:element name="DuplicateShapeResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="DuplicateShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ArrangeShape">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="orderMode" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ArrangeShapeResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ArrangeShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="UngroupShape">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="UngroupShapeResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="UngroupShapeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ApplyShapeStyle">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="styleId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ApplyShapeStyleResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ApplyShapeStyleResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ApplyShapeFill">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="RGBColor" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="themeColor" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

        <xs:element minOccurs="0" name="colorLuminance" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="fTintColor" type="xs:boolean"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ApplyShapeFillResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ApplyShapeFillResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="RemoveShapeFill">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="RemoveShapeFillResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="RemoveShapeFillResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ApplyShapeOutlineColor">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="RGBColor" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="themeColor" type="xs:int"/>
            <xs:element minOccurs="0" name="colorLuminance" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="fTintColor" type="xs:boolean"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ApplyShapeOutlineColorResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ApplyShapeOutlineColorResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ApplyShapeOutlineWidth">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="width" type="xs:double"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ApplyShapeOutlineWidthResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ApplyShapeOutlineWidthResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>

```

```

<xs:element name="ApplyShapeOutlineDashStyle">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="dashType" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ApplyShapeOutlineDashStyleResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ApplyShapeOutlineDashStyleResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ApplyShapeOutlineEndStyle">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="headEndType" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="tailEndType" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ApplyShapeOutlineEndStyleResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ApplyShapeOutlineEndStyleResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="RemoveShapeOutline">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="RemoveShapeOutlineResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="RemoveShapeOutlineResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SaveAndClose">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="fReleaseLock" type="xs:boolean"/>
      <xs:element minOccurs="0" name="fWaitForSave" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SaveAndCloseResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SaveAndCloseResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

    </xs:complexType>
  </xs:element>
  <xs:element name="UpdateCanary">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="UpdateCanaryResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="UpdateCanaryResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Print">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="PrintResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="PrintResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ChangeSmartArtLayout">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="layoutUri" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="category" nillable="true" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ChangeSmartArtLayoutResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ChangeSmartArtLayoutResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ChangeSmartArtColor">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
        <xs:element minOccurs="0" name="colorTransUri" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="category" nillable="true" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ChangeSmartArtColorResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ChangeSmartArtColorResult" nillable="true" type="tns2:ServiceResult"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

</xs:element>
<xs:element name="ChangeSmartArtStyle">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="styleUri" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="category" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ChangeSmartArtStyleResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ChangeSmartArtStyleResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="InsertSmartArt">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="layoutUri" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="category" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="InsertSmartArtResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="InsertSmartArtResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ResetSmartArt">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ResetSmartArtResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ResetSmartArtResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ReverseSmartArt">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ReverseSmartArtResponse">
  <xs:complexType>
    <xs:sequence>

```

```

        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ReverseSmartArtResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ApplyTheme">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="themeFamilyId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="variantIndex" type="xs:int"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ApplyThemeResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ApplyThemeResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetWordArt">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="styleId" type="xs:int"/>
            <xs:element minOccurs="0" name="isClear" type="xs:boolean"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetWordArtResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetWordArtResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="InsertClipart">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="assetId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="Lcid" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="alt" nillable="true" type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="InsertClipartResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="InsertClipartResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="PastePicture">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="url" nillable="true" type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>

```



```

        <xs:element minOccurs="0" name="alt" nillable="true" type="xs:string"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="PastePictureResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="PastePictureResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ShapeFormatPainting">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="sourceSlideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="sourceShapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="destinationSlideId" nillable="true"
type="xs:string"/>
            <xs:element minOccurs="0" name="destinationShapeId" type="xs:unsignedInt"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ShapeFormatPaintingResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ShapeFormatPaintingResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="AddComment">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="text" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="timeZoneBias" type="xs:int"/>
            <xs:element minOccurs="0" name="parentAuthor" type="xs:int"/>
            <xs:element minOccurs="0" name="parentIndex" type="xs:int"/>
            <xs:element minOccurs="0" name="fHasPos" type="xs:boolean"/>
            <xs:element minOccurs="0" name="x" type="xs:int"/>
            <xs:element minOccurs="0" name="y" type="xs:int"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="AddCommentResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="AddCommentResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="DeleteComment">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="authorId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="index" type="xs:unsignedInt"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="DeleteCommentResponse">
    <xs:complexType>
        <xs:sequence>

```

```

        <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="DeleteCommentResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="EditComment">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="text" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="authorId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="index" type="xs:unsignedInt"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="EditCommentResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="EditCommentResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="MoveComment">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="authorId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="index" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="x" type="xs:double"/>
            <xs:element minOccurs="0" name="y" type="xs:double"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="MoveCommentResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="MoveCommentResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetShapeBold">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="isBold" type="xs:boolean"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetShapeBoldResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeBoldResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetShapeItalic">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="isItalic" type="xs:boolean"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>

```

```

        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetShapeItalicResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeItalicResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetShapeUnderline">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="isUnderline" type="xs:boolean"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetShapeUnderlineResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeUnderlineResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetShapeAlignment">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="textAlignment" type="xs:unsignedInt"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetShapeAlignmentResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeAlignmentResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetShapeFontName">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
            <xs:element minOccurs="0" name="fontName" nillable="true" type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetShapeFontNameResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeFontNameResult" nillable="true" type="tns2:ServiceResult"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetShapeFontSize">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>

```

```

    <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="fontSize" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="SetShapeFontSizeResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeFontSizeResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SetShapeFontColor">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="RGBColor" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SetShapeFontColorResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeFontColorResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SetShapeBullet">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="textBullet" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SetShapeBulletResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeBulletResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SetShapeTextDirection">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="shapeId" type="xs:unsignedInt"/>
      <xs:element minOccurs="0" name="textDirection" type="xs:unsignedInt"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SetShapeTextDirectionResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="SetShapeTextDirectionResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ChangeLayout">

```

```
<xs:complexType>
  <xs:sequence>
    <xs:element minOccurs="0" name="presentationId" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="slideId" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="mainMasterId" type="xs:unsignedInt"/>
    <xs:element minOccurs="0" name="contentMasterId" type="xs:unsignedInt"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ChangeLayoutResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element xmlns:tns2="http://schemas.datacontract.org/2004/07/p" minOccurs="0"
name="ChangeLayoutResult" nillable="true" type="tns2:ServiceResult"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```

8 Appendix C: 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 PowerPoint 2010
- Microsoft PowerPoint Online
- Microsoft PowerPoint 2013
- Microsoft PowerPoint 2016

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

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

<1> [Section 2.1](#): PowerPoint Online servers support JSON [\[RFC4627\]](#) over HTTP and HTTPS.

<2> [Section 2.2.4](#): Support for this complex type was added in Microsoft Office 2013.

<3> [Section 2.2.4](#): Support for this complex type was added in Office 2013.

<4> [Section 2.2.4](#): Support for this complex type was added in Office 2013.

<5> [Section 2.2.4](#): Support for this complex type was added in Office 2013.

<6> [Section 2.2.4](#): Support for this complex type was added in Office 2013.

<7> [Section 2.2.4](#): Support for this type was added in Microsoft Office 2010 suites Service Pack 1.

<8> [Section 2.2.4](#): Support for this complex type was added in Office 2013.

<9> [Section 2.2.4](#): Support for this complex type was added in Office 2013.

<10> [Section 2.2.4](#): Support for this complex type was added in Office 2013.

<11> [Section 2.2.4](#): Support for this complex type was added in Office 2013.

<12> [Section 2.2.4](#): Support for this complex type was added in Office 2013.

<13> [Section 2.2.4](#): Support for this complex type was added in Office 2013.

<14> [Section 2.2.4](#): Support for this type was added in Office 2010 Service Pack 1.

<15> [Section 2.2.4](#): Support for this complex type was added in Office 2013.

<16> [Section 2.2.4](#): Support for this type was added in Office 2013.

<17> [Section 2.2.4](#): Support for this complex type was added in Office 2013.

<18> [Section 2.2.4.1](#): Support for this complex type was added in Office 2013.

<19> [Section 2.2.4.3](#): Support for this complex type was added in Office 2013.

<20> [Section 2.2.4.4](#): Support for this complex type was added in Office 2013.

- <21> [Section 2.2.4.12](#): Support for this complex type was added in Office 2013.
- <22> [Section 2.2.4.15](#): Support for this complex type was added in Office 2013.
- <23> [Section 2.2.4.16](#): Support for this type was added in Office 2010 Service Pack 1.
- <24> [Section 2.2.4.17](#): Support for this complex type was added in Office 2013.
- <25> [Section 2.2.4.18](#): Support for this complex type was added in Office 2013.
- <26> [Section 2.2.4.19](#): Support for this complex type was added in Office 2013.
- <27> [Section 2.2.4.22](#): Support for this complex type was added in Office 2013.
- <28> [Section 2.2.4.24](#): Support for this complex type was added in Office 2013.
- <29> [Section 2.2.4.25](#): Support for this element was added in Office 2010 Service Pack 1.
- <30> [Section 2.2.4.26](#): Support for this element was added in Office 2010 Service Pack 1.
- <31> [Section 2.2.4.32](#): Support for this complex type was added in Office 2013.
- <32> [Section 2.2.4.39](#): Support for this type was added in Office 2010 Service Pack 1.
- <33> [Section 2.2.4.40](#): Support for this complex type was added in Office 2013.
- <34> [Section 2.2.4.44](#): Support for this type was added in Office 2013.
- <35> [Section 2.2.4.48](#): Support for this complex type was added in Office 2013.
- <36> [Section 2.2.4.51](#): PowerPoint Online defines this type as a **string** ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the identifier of the image.

To retrieve an image, the protocol client sends a request to the protocol server at the endpoint formed by appending `"/_layouts/editImageHandler.ashx"` to the URL of the website, for example `http://www.contoso.com/Repository/_layouts/editImageHandler.ashx`.

The parameters to the endpoint are specified by the following table:

Parameter name	Definition
Rid	MUST be <code>ImgId</code> as specified by ViewElement (section 2.2.4.51).
Pid	MUST be a valid identifier for a presentation as specified in GetPresentationId (section 3.1.4.27).

<37> [Section 3.1.3](#): The URL conforms to the following structure in Office 2010: *base URL/_vti_bin/ppt/edit.svc*

<38> [Section 3.1.3](#): The insert picture operation is used to insert a picture into a presentation.

In Office 2010, the protocol server endpoint to insert picture content is formed by appending `"/_layouts/pptInsertPicture.ashx"` to the URL of the website, for example `http://www.contoso.com/_layouts/pptInsertPicture.ashx`.

In Office 2013, the protocol server endpoint to insert picture content is formed by appending `"/p/pptInsertPicture.ashx"` to the base URL, for example `http://www.contoso.com/p/pptInsertPicture.ashx`.

The parameters to the endpoint are specified by the following table:

Parameter name	Definition
PresentationId	MUST be a valid identifier of a presentation as specified by section 3.1.4.27.
SlideId	MUST be a valid ST_SlideId ([ISO/IEC29500-1:2011] section 19.7.13) that specifies the identifier of the presentation slide.
ShapeId	MUST be a valid ST_DrawingElementId ([ISO/IEC29500-1:2011] section 20.1.10.21) that specifies the identifier of the shape.

<39> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<40> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<41> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<42> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<43> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<44> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<45> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<46> [Section 3.1.4](#): Support for this operation was added in Office 2010 Service Pack 1.

<47> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<48> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<49> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<50> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<51> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<52> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<53> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<54> [Section 3.1.4](#): Support for this operation was added in Office 2010 Service Pack 1.

<55> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<56> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<57> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<58> [Section 3.1.4](#): Support for this operation was added in Office 2010 Service Pack 1.

<59> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<60> [Section 3.1.4](#): Support for this operation was added in Office 2013.

<61> [Section 3.1.4](#): Support for this operation was added in Office 2013.

[<62> Section 3.1.4](#): Support for this operation was added in Office 2013.

[<63> Section 3.1.4](#): Support for this operation was added in Office 2013.

[<64> Section 3.1.4](#): Support for this operation was added in Office 2013.

[<65> Section 3.1.4](#): Support for this operation was added in Office 2013.

[<66> Section 3.1.4](#): Support for this operation was added in Office 2013.

[<67> Section 3.1.4](#): Support for this operation was added in Office 2013.

[<68> Section 3.1.4](#): Support for this operation was added in Office 2013.

[<69> Section 3.1.4](#): Support for this operation was added in Office 2013.

[<70> Section 3.1.4.1](#): Support for this operation was added in Office 2013.

[<71> Section 3.1.4.2](#): Support for this operation was added in Office 2013.

[<72> Section 3.1.4.3](#): Support for this operation was added in Office 2013.

[<73> Section 3.1.4.4](#): Support for this operation was added in Office 2013.

[<74> Section 3.1.4.5](#): Support for this operation was added in Office 2013.

[<75> Section 3.1.4.6](#): Support for this operation was added in Office 2013.

[<76> Section 3.1.4.7](#): Support for this operation was added in Office 2013.

[<77> Section 3.1.4.8](#): Support for this operation was added in Office 2010 Service Pack 1.

[<78> Section 3.1.4.9](#): Support for this operation was added in Office 2013.

[<79> Section 3.1.4.16](#): Support for this operation was added in Office 2013.

[<80> Section 3.1.4.18](#): Support for this operation was added in Office 2013.

[<81> Section 3.1.4.20](#): Support for this operation was added in Office 2013.

[<82> Section 3.1.4.21](#): Support for this operation was added in Office 2013.

[<83> Section 3.1.4.22](#): Support for this operation was added in Office 2013.

[<84> Section 3.1.4.24](#): Support for this operation was added in Office 2013.

[<85> Section 3.1.4.28](#): Support for this operation was added in Office 2010 Service Pack 1.

[<86> Section 3.1.4.29](#): Support for this operation was added in Office 2013.

[<87> Section 3.1.4.33](#): Support for this operation was added in Office 2013.

[<88> Section 3.1.4.34](#): Support for this operation was added in Office 2013.

[<89> Section 3.1.4.37](#): Support for this operation was added in Office 2010 Service Pack 1.

[<90> Section 3.1.4.39](#): Support for this operation was added in Office 2013.

[<91> Section 3.1.4.40](#): Support for this operation was added in Office 2013.

[<92> Section 3.1.4.41](#): Support for this operation was added in Office 2013.

[<93> Section 3.1.4.44](#): Support for this operation was added in Office 2013.

<94> [Section 3.1.4.46](#): Support for this operation was added in Office 2013.

<95> [Section 3.1.4.48](#): Support for this operation was added in Office 2013.

<96> [Section 3.1.4.50](#): Support for this operation was added in Office 2013.

<97> [Section 3.1.4.60](#): Support for this operation was added in Office 2013.

<98> [Section 3.1.4.61](#): Support for this operation was added in Office 2013.

<99> [Section 3.1.4.63](#): Support for this operation was added in Office 2013.

<100> [Section 3.1.4.66](#): Support for this operation was added in Office 2013.

<101> [Section 5.1](#): PowerPoint Online uses the canary mitigation as follows. On a request to **GetPresentationId** (section 3.1.4.27), an HTTP cookie with a name of X-Key is included in the response. The value of this cookie is then required to be included by subsequent operation calls that might result in a change to a presentation. For any subsequent operation, an additional cookie with the name X-NewKey can also be included which allows the protocol server to inform the protocol client of the new canary value, for example when the canary has expired.

9 Change Tracking

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

10 Index

A

Abstract data model
 [server](#) 61
[Applicability](#) 24
[ArrayOfComment complex type](#) 28
[ArrayOfContentMasterFe complex type](#) 29
[ArrayOfEditAnimInfo complex type](#) 29
[ArrayOfEditorInfo complex type](#) 29
[ArrayOfEditShape complex type](#) 29
[ArrayOfEditSlide complex type](#) 30
[ArrayOfEditSlideChanges complex type](#) 30
[ArrayOfEditSlideInfoFe complex type](#) 30
[ArrayOfMainMasterFe complex type](#) 30
[ArrayOfMainMasterResourcesFe complex type](#) 31
[ArrayOfSlideUpdateInfo complex type](#) 31
[ArrayOfStyleGalleryItem complex type](#) 31
[ArrayOfunsignedInt complex type](#) 32
[ArrayOfViewElement complex type](#) 32
[Attribute groups](#) 59
[Attributes](#) 59

B

[BoundingRegion complex type](#) 32

C

[Capability negotiation](#) 24
[Change tracking](#) 291
[char simple type](#) 53
[ClientActions simple type](#) 53
[ClippingInfo complex type](#) 33
[CoauthState complex type](#) 33
[CoauthUpdate complex type](#) 34
[Comment complex type](#) 34
[Complex types](#) 26
 [ArrayOfComment](#) 28
 [ArrayOfContentMasterFe](#) 29
 [ArrayOfEditAnimInfo](#) 29
 [ArrayOfEditorInfo](#) 29
 [ArrayOfEditShape](#) 29
 [ArrayOfEditSlide](#) 30
 [ArrayOfEditSlideChanges](#) 30
 [ArrayOfEditSlideInfoFe](#) 30
 [ArrayOfMainMasterFe](#) 30
 [ArrayOfMainMasterResourcesFe](#) 31
 [ArrayOfSlideUpdateInfo](#) 31
 [ArrayOfStyleGalleryItem](#) 31
 [ArrayOfunsignedInt](#) 32
 [ArrayOfViewElement](#) 32
 [BoundingRegion](#) 32
 [ClippingInfo](#) 33
 [CoauthState](#) 33
 [CoauthUpdate](#) 34
 [Comment](#) 34
 [ContentMasterFe](#) 35
 [ContentUpdateInfo](#) 35
 [EditAnimInfo](#) 35
 [EditCommandResponse](#) 36
 [EditorInfo](#) 36
 [EditPresentationChanges](#) 37

[EditPresentationInfo](#) 38
 [EditPresentationResources](#) 38
 [EditShape](#) 39
 [EditSlide](#) 41
 [EditSlideChanges](#) 42
 [EditSlideInfoFe](#) 43
 [EditTransitionInfo](#) 44
 [FontSizeMapping](#) 44
 [Html](#) 45
 [MainMasterFe](#) 45
 [MainMasterResourcesFe](#) 46
 [Notes](#) 46
 [Point](#) 46
 [PptViewingService.PrintResult](#) 47
 [PresetSchemeColorMapping](#) 47
 [Rectangle](#) 48
 [ServiceError](#) 48
 [ServiceResult](#) 49
 [SlideID](#) 49
 [SlideIdList](#) 49
 [SlideUpdateInfo](#) 50
 [SmartArtInfo](#) 50
 [StyleGalleryItem](#) 50
 [Theme](#) 51
 [ThemeColor](#) 51
 [ViewElement](#) 52
[ContentMasterFe complex type](#) 35
[ContentUpdateInfo complex type](#) 35

D

Data model - abstract
 [server](#) 61
[duration simple type](#) 54

E

[EditAnimInfo complex type](#) 35
[EditCommandResponse complex type](#) 36
[EditorInfo complex type](#) 36
[EditPresentationChanges complex type](#) 37
[EditPresentationInfo complex type](#) 38
[EditPresentationResources complex type](#) 38
[EditShape complex type](#) 39
[EditSlide complex type](#) 41
[EditSlideChanges complex type](#) 42
[EditSlideInfoFe complex type](#) 43
[EditTransitionInfo complex type](#) 44
[ErrorCode simple type](#) 54
Events
 [local - server](#) 217
 [timer - server](#) 217
Examples
 [overview](#) 218
 [sample protocol interaction](#) 218

F

[Fields - vendor-extensible](#) 24
[FontSizeMapping complex type](#) 44
[Full WSDL](#) 229
[Full XML schema](#) 256

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared.Schema> 256

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.Interface.Schema> 257

<http://schemas.datacontract.org/2004/07/pSchema> 258

<http://schemas.microsoft.com/2003/10/Serialization/Schema> 266

<http://schemas.microsoft.com/2003/10/Serialization/Arrays.Schema> 266

<http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/Schema> 267

G

[Glossary](#) 20
[Groups](#) 59
[guid simple type](#) 58

H

[Html complex type](#) 45

I

[Implementer - security considerations](#) 228
[Index of security parameters](#) 228
[Informative references](#) 23
Initialization
 [server](#) 61
[Introduction](#) 20

L

Local events
 [server](#) 217

M

[MainMasterFe complex type](#) 45
[MainMasterResourcesFe complex type](#) 46
Message processing
 [server](#) 61
Messages
 [ArrayOfComment complex type](#) 28
 [ArrayOfContentMasterFe complex type](#) 29
 [ArrayOfEditAnimInfo complex type](#) 29
 [ArrayOfEditorInfo complex type](#) 29
 [ArrayOfEditShape complex type](#) 29
 [ArrayOfEditSlide complex type](#) 30
 [ArrayOfEditSlideChanges complex type](#) 30
 [ArrayOfEditSlideInfoFe complex type](#) 30
 [ArrayOfMainMasterFe complex type](#) 30
 [ArrayOfMainMasterResourcesFe complex type](#) 31
 [ArrayOfSlideUpdateInfo complex type](#) 31
 [ArrayOfStyleGalleryItem complex type](#) 31

[ArrayOfunsignedInt complex type](#) 32
[ArrayOfViewElement complex type](#) 32
[attribute groups](#) 59
[attributes](#) 59
[BoundingRegion complex type](#) 32
[char simple type](#) 53
[ClientActions simple type](#) 53
[ClippingInfo complex type](#) 33
[CoauthState complex type](#) 33
[CoauthUpdate complex type](#) 34
[Comment complex type](#) 34
[complex types](#) 26
[ContentMasterFe complex type](#) 35
[ContentUpdateInfo complex type](#) 35
[duration simple type](#) 54
[EditAnimInfo complex type](#) 35
[EditCommandResponse complex type](#) 36
[EditorInfo complex type](#) 36
[EditPresentationChanges complex type](#) 37
[EditPresentationInfo complex type](#) 38
[EditPresentationResources complex type](#) 38
[EditShape complex type](#) 39
[EditSlide complex type](#) 41
[EditSlideChanges complex type](#) 42
[EditSlideInfoFe complex type](#) 43
[EditTransitionInfo complex type](#) 44
[elements](#) 26
[enumerated](#) 26
[ErrorCode simple type](#) 54
[FontSizeMapping complex type](#) 44
[groups](#) 59
[guid simple type](#) 58
[Html complex type](#) 45
[MainMasterFe complex type](#) 45
[MainMasterResourcesFe complex type](#) 46
[namespaces](#) 25
[Notes complex type](#) 46
[Point complex type](#) 46
[PptViewingService.PrintResult complex type](#) 47
[PresetSchemeColorMapping complex type](#) 47
[Rectangle complex type](#) 48
[ServiceError complex type](#) 48
[ServiceErrorType simple type](#) 59
[ServiceResult complex type](#) 49
[simple types](#) 53
[SlideID complex type](#) 49
[SlideIdList complex type](#) 49
[SlideUpdateInfo complex type](#) 50
[SmartArtInfo complex type](#) 50
[StyleGalleryItem complex type](#) 50
[syntax](#) 25
[Theme complex type](#) 51
[ThemeColor complex type](#) 51
[transport](#) 25
[ViewElement complex type](#) 52

N

[Namespaces](#) 25
[Normative references](#) 22
[Notes complex type](#) 46

O

Operations

[AddComment](#) 64
[ApplyShapeFill](#) 67
[ApplyShapeOutlineColor](#) 69
[ApplyShapeOutlineDashStyle](#) 72
[ApplyShapeOutlineEndStyle](#) 74
[ApplyShapeOutlineWidth](#) 77
[ApplyShapeStyle](#) 80
[ApplyTheme](#) 82
[ArrangeShape](#) 84
[ChangeLayout](#) 87
[ChangePictureStyle](#) 89
[ChangeSmartArtColor](#) 91
[ChangeSmartArtLayout](#) 94
[ChangeSmartArtStyle](#) 96
[ClearPlaceholder](#) 98
[DeleteComment](#) 100
[DeleteSlide](#) 103
[DuplicateShape](#) 105
[DuplicateSlide](#) 107
[EditComment](#) 109
[FlipShape](#) 112
[GetCoauthUpdates](#) 114
[GetEditPresInfo](#) 116
[GetEditPresInfoGetEditSlideById](#) 118
[GetEditPresResources](#) 121
[GetEditSlide](#) 123
[GetPresentationId](#) 125
[InsertClipart](#) 127
[InsertShape](#) 129
[InsertSlide](#) 132
[InsertSmartArt](#) 134
[LogULS](#) 136
[MoveComment](#) 138
[MoveShape](#) 141
[MoveSlide](#) 143
[PastePicture](#) 145
[Print](#) 147
[Redo](#) 149
[RemoveShapeFill](#) 151
[RemoveShapeOutline](#) 154
[ReorderAnimation](#) 156
[ReplaceNotes](#) 158
[ReplaceText](#) 160
[ResetPicture](#) 162
[ResetSmartArt](#) 165
[ResizeShape](#) 167
[ReverseSmartArt](#) 169
[RotateShape](#) 171
[SaveAndClose](#) 174
[SetAnimation](#) 176
[SetShapeAlignment](#) 179
[SetShapeBold](#) 181
[SetShapeBullet](#) 183
[SetShapeFontColor](#) 186
[SetShapeFontName](#) 188
[SetShapeFontSize](#) 190
[SetShapeItalic](#) 193
[SetShapeTextDirection](#) 195
[SetShapeUnderline](#) 197
[SetTransition](#) 199
[SetTransitionApplyAll](#) 202
[SetWordArt](#) 204
[ShapeFormatPainting](#) 207
[ShowHideSlide](#) 209
[Undo](#) 211

[UngroupShape](#) 213
[UpdateCanary](#) 215
[Overview \(synopsis\)](#) 23

P

[Parameters - security index](#) 228
[Point complex type](#) 46
[PptViewingService.PrintResult complex type](#) 47
[Preconditions](#) 23
[Prerequisites](#) 23
[PresetSchemeColorMapping complex type](#) 47
[Product behavior](#) 286
 Protocol Details
 [overview](#) 60

R

[Rectangle complex type](#) 48
[References](#) 22
 [informative](#) 23
 [normative](#) 22
[Relationship to other protocols](#) 23

S

[Sample protocol interaction example](#) 218
 Security
 [implementer considerations](#) 228
 [parameter index](#) 228
 Sequencing rules
 [server](#) 61
 Server
 [abstract data model](#) 61
 [AddComment operation](#) 64
 [ApplyShapeFill operation](#) 67
 [ApplyShapeOutlineColor operation](#) 69
 [ApplyShapeOutlineDashStyle operation](#) 72
 [ApplyShapeOutlineEndStyle operation](#) 74
 [ApplyShapeOutlineWidth operation](#) 77
 [ApplyShapeStyle operation](#) 80
 [ApplyTheme operation](#) 82
 [ArrangeShape operation](#) 84
 [ChangeLayout operation](#) 87
 [ChangePictureStyle operation](#) 89
 [ChangeSmartArtColor operation](#) 91
 [ChangeSmartArtLayout operation](#) 94
 [ChangeSmartArtStyle operation](#) 96
 [ClearPlaceholder operation](#) 98
 [DeleteComment operation](#) 100
 [DeleteSlide operation](#) 103
 [details](#) 60
 [DuplicateShape operation](#) 105
 [DuplicateSlide operation](#) 107
 [EditComment operation](#) 109
 [FlipShape operation](#) 112
 [GetCoauthUpdates operation](#) 114
 [GetEditPresInfo operation](#) 116
 [GetEditPresInfoGetEditSlideById operation](#) 118
 [GetEditPresResources operation](#) 121
 [GetEditSlide operation](#) 123
 [GetPresentationId operation](#) 125
 [initialization](#) 61
 [InsertClipart operation](#) 127
 [InsertShape operation](#) 129

- [InsertSlide operation](#) 132
- [InsertSmartArt operation](#) 134
- [local events](#) 217
- [LogULS operation](#) 136
- [message processing](#) 61
- [MoveComment operation](#) 138
- [MoveShape operation](#) 141
- [MoveSlide operation](#) 143
- [PastePicture operation](#) 145
- [Print operation](#) 147
- [Redo operation](#) 149
- [RemoveShapeFill operation](#) 151
- [RemoveShapeOutline operation](#) 154
- [ReorderAnimation operation](#) 156
- [ReplaceNotes operation](#) 158
- [ReplaceText operation](#) 160
- [ResetPicture operation](#) 162
- [ResetSmartArt operation](#) 165
- [ResizeShape operation](#) 167
- [ReverseSmartArt operation](#) 169
- [RotateShape operation](#) 171
- [SaveAndClose operation](#) 174
- [sequencing rules](#) 61
- [SetAnimation operation](#) 176
- [SetShapeAlignment operation](#) 179
- [SetShapeBold operation](#) 181
- [SetShapeBullet operation](#) 183
- [SetShapeFontColor operation](#) 186
- [SetShapeFontName operation](#) 188
- [SetShapeFontSize operation](#) 190
- [SetShapeItalic operation](#) 193
- [SetShapeTextDirection operation](#) 195
- [SetShapeUnderline operation](#) 197
- [SetTransition operation](#) 199
- [SetTransitionApplyAll operation](#) 202
- [SetWordArt operation](#) 204
- [ShapeFormatPainting operation](#) 207
- [ShowHideSlide operation](#) 209
- [timer events](#) 217
- [timers](#) 61
- [Undo operation](#) 211
- [UngroupShape operation](#) 213
- [UpdateCanary operation](#) 215
- [ServiceError complex type](#) 48
- [ServiceErrorType simple type](#) 59
- [ServiceResult complex type](#) 49
- [Simple types](#) 53
 - [char](#) 53
 - [ClientActions](#) 53
 - [duration](#) 54
 - [ErrorCode](#) 54
 - [guid](#) 58
 - [ServiceErrorType](#) 59
- [SlideID complex type](#) 49
- [SlideIdList complex type](#) 49
- [SlideUpdateInfo complex type](#) 50
- [SmartArtInfo complex type](#) 50
- [Standards assignments](#) 24
- [StyleGalleryItem complex type](#) 50
- Syntax
 - [messages - overview](#) 25

T

- [Theme complex type](#) 51

- [ThemeColor complex type](#) 51
- Timer events
 - [server](#) 217
- Timers
 - [server](#) 61
- [Tracking changes](#) 291
- [Transport](#) 25
- Types
 - [complex](#) 26
 - [simple](#) 53

V

- [Vendor-extensible fields](#) 24
- [Versioning](#) 24
- [ViewElement complex type](#) 52

W

- [WSDL](#) 229

X

- [XML schema](#) 256

- <http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Interface.Shared.Schema> 256

- <http://schemas.datacontract.org/2004/07/Microsoft.Office.Server.Powerpoint.Pipe.Interface.Schema> 257

- <http://schemas.datacontract.org/2004/07/pSchema> 258

- <http://schemas.microsoft.com/2003/10/Serialization/Schema> 266

- <http://schemas.microsoft.com/2003/10/Serialization/Arrays.Schema> 266

- <http://schemas.microsoft.com/office/PowerPoint/Server/WebServices/PowerPointEditServerInternalService/Schema> 267