

CSTA

CSTADLL
Version 2.2.3
CSTADLL
Reference Manual

The software described in this document is furnished under a license agreement and may be used only in accordance with the terms of this agreement.

Copyright Notice

Copyright ©1997-2015 Objective Systems, Inc. All rights reserved.

This document may be distributed in any form, electronic or otherwise, provided that it is distributed in its entirety and that the copyright and this notice are included.

Author's Contact Information

Comments, suggestions, and inquiries regarding CSTADLL may be submitted via electronic mail to info@obj-sys.com.

Contents

1	CSTADLL	1
2	Namespace Documentation	3
2.1	Package Com.Objsys.Csta.Common	3
2.1.1	Detailed Description	3
2.2	Package Com.Objsys.Csta.Devices	4
2.2.1	Detailed Description	4
2.3	Package Com.Objsys.Csta.Phase1	5
2.3.1	Detailed Description	5
2.4	Package Com.Objsys.Csta.Phase2	6
2.4.1	Detailed Description	6
2.5	Package Com.Objsys.Csta.Phase3	7
2.5.1	Detailed Description	7
2.6	Package Com.Objsys.Csta.Xml	8
2.6.1	Detailed Description	8
3	Class Documentation	9
3.1	Alcatel4400 Class Reference	9
3.1.1	Detailed Description	9
3.1.2	Constructor & Destructor Documentation	9
3.1.2.1	Alcatel4400	9
3.1.2.2	Alcatel4400	10
3.1.3	Member Function Documentation	10
3.1.3.1	EncodeACSEConnectionRequest	10
3.1.3.2	EncodeMakeCallRequest	10
3.1.3.3	MakeACSEAssociation	10
3.2	AlcatelOXE Class Reference	11
3.2.1	Detailed Description	11
3.2.2	Constructor & Destructor Documentation	11

3.2.2.1	AlcatelOXE	11
3.2.2.2	AlcatelOXE	11
3.3	AlcatelOXO Class Reference	12
3.3.1	Detailed Description	12
3.3.2	Constructor & Destructor Documentation	12
3.3.2.1	AlcatelOXO	12
3.3.2.2	AlcatelOXO	12
3.3.3	Member Function Documentation	12
3.3.3.1	EncodeACSEConnectionRequest	12
3.3.3.2	MakeACSEAssociation	13
3.4	CSTAContext Class Reference	14
3.4.1	Detailed Description	14
3.4.2	Property Documentation	14
3.4.2.1	ResponseFromPBX	14
3.4.2.2	ResponsesFromPBX	14
3.4.2.3	XMLResponseFromPBX	14
3.4.2.4	XMLResponsesFromPBX	14
3.5	CSTAResponseInfo Class Reference	15
3.5.1	Detailed Description	15
3.5.2	Property Documentation	15
3.5.2.1	ResponseFromPBX	15
3.5.2.2	ResponsesFromPBX	15
3.5.2.3	StatusCode	15
3.5.2.4	StatusMessage	15
3.5.2.5	XMLResponseFromPBX	15
3.5.2.6	XMLResponsesFromPBX	16
3.6	GenericCSTAp1 Class Reference	17
3.6.1	Detailed Description	18
3.6.2	Constructor & Destructor Documentation	18
3.6.2.1	GenericCSTAp1	18
3.6.2.2	GenericCSTAp1	19
3.6.3	Member Function Documentation	19
3.6.3.1	AnswerCall	19
3.6.3.2	AnswerCall	19
3.6.3.3	AnswerCall	19
3.6.3.4	ClearConnection	19
3.6.3.5	ClearDoNotDisturb	20

3.6.3.6	ClearMessageWaiting	20
3.6.3.7	ConsultationCall	20
3.6.3.8	DivertCall	20
3.6.3.9	EncodeACSEConnectionRequest	21
3.6.3.10	EncodeACSEReleaseRequest	21
3.6.3.11	EncodeAnswerCallRequest	21
3.6.3.12	EncodeAnswerCallRequest	22
3.6.3.13	EncodeAnswerCallRequest	22
3.6.3.14	EncodeClearConnectionRequest	22
3.6.3.15	EncodeConsultationCallRequest	22
3.6.3.16	EncodeDivertCallRequest	23
3.6.3.17	EncodeMakeCallRequest	23
3.6.3.18	EncodeMonitorStartRequest	23
3.6.3.19	EncodeMonitorStartRequest	24
3.6.3.20	EncodeMonitorStopRequest	24
3.6.3.21	EncodeMonitorStopRequest	24
3.6.3.22	EncodeQueryDeviceRequest	25
3.6.3.23	EncodeRetrieveCallRequest	25
3.6.3.24	EncodeROSERequestHeader	25
3.6.3.25	EncodeSetOrClearDNDRequest	25
3.6.3.26	EncodeSetOrClearMWRequest	26
3.6.3.27	EncodeSnapshotDeviceRequest	26
3.6.3.28	EncodeTransferCallRequest	26
3.6.3.29	EncodeTransferCallRequest	27
3.6.3.30	MakeACSEAssociation	27
3.6.3.31	MakeCall	27
3.6.3.32	MonitorStart	28
3.6.3.33	MonitorStart	28
3.6.3.34	MonitorStop	28
3.6.3.35	MonitorStop	28
3.6.3.36	MonitorStop	29
3.6.3.37	QueryDevice	29
3.6.3.38	ReleaseACSEAssociation	29
3.6.3.39	RetrieveCall	29
3.6.3.40	SetDoNotDisturb	29
3.6.3.41	SetMessageWaiting	30
3.6.3.42	SnapshotDevice	30

3.6.3.43	TransferCall	30
3.6.3.44	TransferCall	30
3.6.4	Property Documentation	31
3.6.4.1	SessionObject	31
3.6.4.2	ThreadContext	31
3.7	GenericCSTAp2 Class Reference	32
3.7.1	Detailed Description	33
3.7.2	Constructor & Destructor Documentation	33
3.7.2.1	GenericCSTAp2	33
3.7.2.2	GenericCSTAp2	34
3.7.3	Member Function Documentation	34
3.7.3.1	AnswerCall	34
3.7.3.2	AnswerCall	34
3.7.3.3	AnswerCall	34
3.7.3.4	ClearConnection	34
3.7.3.5	ClearDoNotDisturb	35
3.7.3.6	ClearMessageWaiting	35
3.7.3.7	ConsultationCall	35
3.7.3.8	DivertCall	35
3.7.3.9	EncodeACSEConnectionRequest	36
3.7.3.10	EncodeACSEReleaseRequest	36
3.7.3.11	EncodeAnswerCallRequest	36
3.7.3.12	EncodeAnswerCallRequest	37
3.7.3.13	EncodeAnswerCallRequest	37
3.7.3.14	EncodeClearConnectionRequest	37
3.7.3.15	EncodeConsultationCallRequest	37
3.7.3.16	EncodeDivertCallRequest	38
3.7.3.17	EncodeMakeCallRequest	38
3.7.3.18	EncodeMonitorStartRequest	38
3.7.3.19	EncodeMonitorStartRequest	39
3.7.3.20	EncodeMonitorStopRequest	39
3.7.3.21	EncodeMonitorStopRequest	39
3.7.3.22	EncodeQueryDeviceRequest	40
3.7.3.23	EncodeRetrieveCallRequest	40
3.7.3.24	EncodeROSERequestHeader	40
3.7.3.25	EncodeSetOrClearDNDRequest	41
3.7.3.26	EncodeSetOrClearMWRRequest	41

3.7.3.27	EncodeSnapshotDeviceRequest	41
3.7.3.28	EncodeTransferCallRequest	42
3.7.3.29	EncodeTransferCallRequest	42
3.7.3.30	MakeACSEAssociation	42
3.7.3.31	MakeCall	42
3.7.3.32	MonitorStart	43
3.7.3.33	MonitorStart	43
3.7.3.34	MonitorStop	43
3.7.3.35	MonitorStop	43
3.7.3.36	MonitorStop	44
3.7.3.37	QueryDevice	44
3.7.3.38	ReleaseACSEAssociation	44
3.7.3.39	RetrieveCall	44
3.7.3.40	SetDoNotDisturb	45
3.7.3.41	SetMessageWaiting	45
3.7.3.42	SnapshotDevice	45
3.7.3.43	TransferCall	45
3.7.3.44	TransferCall	46
3.7.4	Property Documentation	46
3.7.4.1	SessionObject	46
3.7.4.2	ThreadContext	46
3.8	GenericCSTAp3 Class Reference	47
3.8.1	Detailed Description	49
3.8.2	Constructor & Destructor Documentation	49
3.8.2.1	GenericCSTAp3	49
3.8.2.2	GenericCSTAp3	49
3.8.3	Member Function Documentation	50
3.8.3.1	AlternateCall	50
3.8.3.2	AnswerCall	50
3.8.3.3	AnswerCall	50
3.8.3.4	AnswerCall	50
3.8.3.5	ClearConnection	51
3.8.3.6	ClearDoNotDisturb	51
3.8.3.7	ClearMessageWaiting	51
3.8.3.8	ConferenceCall	51
3.8.3.9	ConsultationCall	52
3.8.3.10	EncodeACSEConnectionRequest	52

3.8.3.11	EncodeACSEReleaseRequest	52
3.8.3.12	EncodeAlternateCallRequest	52
3.8.3.13	EncodeAnswerCallRequest	53
3.8.3.14	EncodeAnswerCallRequest	53
3.8.3.15	EncodeAnswerCallRequest	53
3.8.3.16	EncodeClearConnectionRequest	54
3.8.3.17	EncodeConferenceCallRequest	54
3.8.3.18	EncodeConsultationCallRequest	54
3.8.3.19	EncodeGetAgentStateRequest	55
3.8.3.20	EncodeGetDNDRquest	55
3.8.3.21	EncodeGetLogicalDevInfoRequest	55
3.8.3.22	EncodeGetPhysicalDevInfoRequest	55
3.8.3.23	EncodeGetSFDevicesRequest	56
3.8.3.24	EncodeHoldCallRequest	56
3.8.3.25	EncodeMakeCallRequest	56
3.8.3.26	EncodeMonitorStartRequest	57
3.8.3.27	EncodeMonitorStartRequest	57
3.8.3.28	EncodeMonitorStopRequest	57
3.8.3.29	EncodeMonitorStopRequest	58
3.8.3.30	EncodeRetrieveCallRequest	58
3.8.3.31	EncodeROSERquestHeader	58
3.8.3.32	EncodeSendDataRequest	59
3.8.3.33	EncodeSetAgentStateRequest	59
3.8.3.34	EncodeSetAgentStateRequest	59
3.8.3.35	EncodeSetDisplayRequest	60
3.8.3.36	EncodeSetOrClearDNDRquest	60
3.8.3.37	EncodeSetOrClearMWRquest	60
3.8.3.38	EncodeSetRingerStatusRequest	61
3.8.3.39	EncodeSingleStepTransferRequest	61
3.8.3.40	EncodeSingleStepTransferRequest	61
3.8.3.41	EncodeSnapshotCallRequest	62
3.8.3.42	EncodeSnapshotDeviceRequest	62
3.8.3.43	EncodeStartDataPathRequest	62
3.8.3.44	EncodeStopDataPathRequest	62
3.8.3.45	EncodeTransferCallRequest	63
3.8.3.46	EncodeTransferCallRequest	63
3.8.3.47	GetAgentState	63

3.8.3.48	GetDoNotDisturb	64
3.8.3.49	GetLogicalDevInfo	64
3.8.3.50	GetPhysicalDevInfo	64
3.8.3.51	GetSFDevices	64
3.8.3.52	GetSFDevices	65
3.8.3.53	HoldCall	65
3.8.3.54	MakeACSEAssociation	65
3.8.3.55	MakeCall	65
3.8.3.56	MonitorStart	65
3.8.3.57	MonitorStart	66
3.8.3.58	MonitorStop	66
3.8.3.59	MonitorStop	66
3.8.3.60	MonitorStop	66
3.8.3.61	ReleaseACSEAssociation	67
3.8.3.62	RetrieveCall	67
3.8.3.63	RingDevice	67
3.8.3.64	SendData	67
3.8.3.65	SetAgentState	68
3.8.3.66	SetAgentState	68
3.8.3.67	SetDisplay	68
3.8.3.68	SetDoNotDisturb	68
3.8.3.69	SetMessageWaiting	69
3.8.3.70	SingleStepTransfer	69
3.8.3.71	SingleStepTransfer	69
3.8.3.72	SnapshotCall	69
3.8.3.73	SnapshotDevice	70
3.8.3.74	StartDataPath	70
3.8.3.75	StopDataPath	70
3.8.3.76	StopRing	70
3.8.3.77	TransferCall	71
3.8.3.78	TransferCall	71
3.8.4	Property Documentation	71
3.8.4.1	SessionObject	71
3.8.4.2	ThreadContext	71
3.9	GenericXML Class Reference	72
3.9.1	Detailed Description	73
3.9.2	Constructor & Destructor Documentation	74

3.9.2.1	GenericXML	74
3.9.2.2	GenericXML	74
3.9.3	Member Function Documentation	74
3.9.3.1	AnswerCall	74
3.9.3.2	AnswerCall	74
3.9.3.3	ClearConnection	74
3.9.3.4	ClearDoNotDisturb	75
3.9.3.5	ClearMessageWaiting	75
3.9.3.6	ConsultationCall	75
3.9.3.7	EncodeAnswerCall	75
3.9.3.8	EncodeAnswerCall	76
3.9.3.9	EncodeClearConnection	76
3.9.3.10	EncodeConsultationCall	76
3.9.3.11	EncodeGetAgentState	77
3.9.3.12	EncodeGetDND	77
3.9.3.13	EncodeGetSFDevices	77
3.9.3.14	EncodeHoldCall	77
3.9.3.15	EncodeInvokeID	78
3.9.3.16	EncodeMakeCall	78
3.9.3.17	EncodeMonitorStart	78
3.9.3.18	EncodeMonitorStart	78
3.9.3.19	EncodeMonitorStop	79
3.9.3.20	EncodeRetrieveCall	79
3.9.3.21	EncodeSendData	79
3.9.3.22	EncodeSetAgentState	79
3.9.3.23	EncodeSetAgentState	80
3.9.3.24	EncodeSetDisplay	80
3.9.3.25	EncodeSetDND	80
3.9.3.26	EncodeSetMsgWaiting	81
3.9.3.27	EncodeSetRingerStatus	81
3.9.3.28	EncodeSingleStepTransfer	81
3.9.3.29	EncodeSnapshotCall	82
3.9.3.30	EncodeSnapshotDevice	82
3.9.3.31	EncodeStartDataPath	82
3.9.3.32	EncodeStartSession	82
3.9.3.33	EncodeStopDataPath	83
3.9.3.34	EncodeStopSession	83

3.9.3.35	EncodeTransferCall	83
3.9.3.36	GetAgentState	83
3.9.3.37	GetDoNotDisturb	84
3.9.3.38	GetSFDevices	84
3.9.3.39	GetSFDevices	84
3.9.3.40	HoldCall	84
3.9.3.41	MakeCall	84
3.9.3.42	MonitorStart	85
3.9.3.43	MonitorStart	85
3.9.3.44	MonitorStop	85
3.9.3.45	MonitorStopAtDevice	85
3.9.3.46	RetrieveCall	86
3.9.3.47	RingDevice	86
3.9.3.48	SendData	86
3.9.3.49	SetAgentState	86
3.9.3.50	SetAgentState	87
3.9.3.51	SetDisplay	87
3.9.3.52	SetDoNotDisturb	87
3.9.3.53	SetMessageWaiting	87
3.9.3.54	SingleStepTransfer	88
3.9.3.55	SnapshotCall	88
3.9.3.56	SnapshotDevice	88
3.9.3.57	StartDataPath	88
3.9.3.58	StartSession	89
3.9.3.59	StartSession	89
3.9.3.60	StopDataPath	89
3.9.3.61	StopRing	89
3.9.3.62	StopSession	89
3.9.3.63	TransferCall	90
3.9.4	Property Documentation	90
3.9.4.1	SessionObject	90
3.9.4.2	ThreadContext	90
3.10	IETF_CSTAp1 Class Reference	91
3.10.1	Detailed Description	91
3.10.2	Constructor & Destructor Documentation	91
3.10.2.1	IETF_CSTAp1	91
3.10.2.2	IETF_CSTAp1	91

3.11 IETF_CSTAp2 Class Reference	92
3.11.1 Detailed Description	92
3.11.2 Constructor & Destructor Documentation	92
3.11.2.1 IETF_CSTAp2	92
3.11.2.2 IETF_CSTAp2	92
3.12 IETF_CSTAp3 Class Reference	93
3.12.1 Detailed Description	93
3.12.2 Constructor & Destructor Documentation	93
3.12.2.1 IETF_CSTAp3	93
3.12.2.2 IETF_CSTAp3	93
3.13 LicenseException Class Reference	94
3.13.1 Detailed Description	94
3.14 PanasonicKXTDA Class Reference	95
3.14.1 Detailed Description	95
3.14.2 Constructor & Destructor Documentation	95
3.14.2.1 PanasonicKXTDA	95
3.14.2.2 PanasonicKXTDA	95
3.15 PanasonicKXTDE Class Reference	96
3.15.1 Detailed Description	96
3.15.2 Member Enumeration Documentation	96
3.15.2.1 DeviceDataTypes	96
3.15.3 Constructor & Destructor Documentation	97
3.15.3.1 PanasonicKXTDE	97
3.15.3.2 PanasonicKXTDE	97
3.15.4 Member Function Documentation	97
3.15.4.1 AcquireControlRight	97
3.15.4.2 ClearMessageWaiting	97
3.15.4.3 EncodeACSEConnectionRequest	97
3.15.4.4 EncodeMakeCallRequest	98
3.15.4.5 EncodeSetDisplayRequest	98
3.15.4.6 GetDeviceData	98
3.15.4.7 GetGroupMembers	99
3.15.4.8 GetSFDevices	99
3.15.4.9 PDFStart	99
3.15.4.10 PDFStop	99
3.15.4.11 ReleaseControlRight	100
3.15.4.12 ResetDisplay	100

3.15.4.13	SendKmeMessage	100
3.15.4.14	SetMessageWaiting	100
3.16	PanasonicNCP Class Reference	101
3.16.1	Detailed Description	101
3.16.2	Constructor & Destructor Documentation	101
3.16.2.1	PanasonicNCP	101
3.16.2.2	PanasonicNCP	101
3.17	PBXSession Class Reference	102
3.17.1	Detailed Description	102
3.17.2	Constructor & Destructor Documentation	103
3.17.2.1	PBXSession	103
3.17.3	Member Function Documentation	103
3.17.3.1	AsyncCallback	103
3.17.3.2	AsyncExceptionCallback	103
3.17.3.3	Close	103
3.17.3.4	ConnectionCallback	103
3.17.3.5	Open	104
3.17.3.6	SendACSEMessage	104
3.17.3.7	SendMessage	104
3.17.3.8	SendMessage	104
3.17.3.9	SendXMLMessage	105
3.17.3.10	SendXMLMessage	105
3.17.3.11	SendXMLSession	105
3.17.3.12	WaitForROSEResponse	105
3.17.3.13	WaitForXMLResponse	105
3.17.3.14	XMLAsyncCallback	106
3.17.4	Property Documentation	106
3.17.4.1	ClientCallback	106
3.17.4.2	ConnectionLostCallback	106
3.17.4.3	DebugClientCallback	106
3.17.4.4	DebugMode	106
3.17.4.5	ExceptionCallback	106
3.17.4.6	MaxReceiveTimeout	106
3.17.4.7	MessageEncoding	106
3.17.4.8	PBXSystem	107
3.17.4.9	Port	107
3.17.4.10	SystemStatusCallback	107

3.17.4.11 XMLClientCallback	107
3.17.4.12 XMLSystemStatusCallback	107
3.18 PBXSessionException Class Reference	108
3.18.1 Detailed Description	108
3.19 PBXSessionHelper Class Reference	109
3.19.1 Detailed Description	109
3.19.2 Property Documentation	109
3.19.2.1 LoggingFolder	109
3.20 Phase1Opcodes Class Reference	110
3.20.1 Detailed Description	110
3.20.2 Member Enumeration Documentation	110
3.20.2.1 Opcodes	110
3.21 Phase2Opcodes Class Reference	111
3.21.1 Detailed Description	111
3.21.2 Member Enumeration Documentation	111
3.21.2.1 Opcodes	111
3.22 Phase3Opcodes Class Reference	112
3.22.1 Detailed Description	112
3.22.2 Member Enumeration Documentation	112
3.22.2.1 Opcodes	112
3.23 PhilipsSopho Class Reference	113
3.23.1 Detailed Description	113
3.23.2 Constructor & Destructor Documentation	113
3.23.2.1 PhilipsSopho	113
3.23.2.2 PhilipsSopho	113
3.23.3 Member Function Documentation	113
3.23.3.1 MakeACSEAssociation	113
3.23.3.2 MakeACSEAssociation	114
3.24 ResetSessionInfo Class Reference	115
3.24.1 Detailed Description	115
3.25 ROSEParseInfo Class Reference	116
3.25.1 Detailed Description	116
3.26 SiemensCap Class Reference	117
3.26.1 Detailed Description	117
3.26.2 Constructor & Destructor Documentation	117
3.26.2.1 SiemensCap	117
3.26.2.2 SiemensCap	117

3.26.3	Member Function Documentation	117
3.26.3.1	EncodeACSEConnectionRequest	117
3.26.3.2	MakeACSEAssociation	118
3.26.3.3	MakeACSEAssociation	118
3.26.3.4	MakeACSEAssociation	118
3.27	SiemensHicom300 Class Reference	120
3.27.1	Detailed Description	120
3.27.2	Constructor & Destructor Documentation	120
3.27.2.1	SiemensHicom300	120
3.27.2.2	SiemensHicom300	120
3.28	SiemensHipath3000p2 Class Reference	121
3.28.1	Detailed Description	121
3.28.2	Constructor & Destructor Documentation	121
3.28.2.1	SiemensHipath3000p2	121
3.28.2.2	SiemensHipath3000p2	121
3.29	SiemensHipath3000p3 Class Reference	122
3.29.1	Detailed Description	122
3.29.2	Constructor & Destructor Documentation	122
3.29.2.1	SiemensHipath3000p3	122
3.29.2.2	SiemensHipath3000p3	122
3.29.3	Member Function Documentation	122
3.29.3.1	EncodeACSEConnectionRequest	122
3.29.3.2	EncodeSingleStepTransferRequest	123
3.29.3.3	MakeACSEAssociation	123
3.30	SiemensHipath4000 Class Reference	124
3.30.1	Detailed Description	124
3.30.2	Constructor & Destructor Documentation	124
3.30.2.1	SiemensHipath4000	124
3.30.2.2	SiemensHipath4000	124
3.31	SiemensRealitis Class Reference	125
3.31.1	Detailed Description	125
3.31.2	Constructor & Destructor Documentation	125
3.31.2.1	SiemensRealitis	125
3.31.2.2	SiemensRealitis	125
3.32	SingleStepTransferInfo Class Reference	126
3.32.1	Detailed Description	126
3.32.2	Property Documentation	126

3.32.2.1	TransferFromDevice	126
3.32.2.2	TransferringCallID	126
3.32.2.3	TransferToDevice	126
3.33	SocketState Class Reference	127
3.33.1	Detailed Description	127
3.33.2	Property Documentation	127
3.33.2.1	AckBuffer	127
3.33.2.2	ReadBuffer	127
3.33.2.3	ReadBuffers	127
3.33.2.4	TotalLength	127
3.34	TadiranCoral Class Reference	128
3.34.1	Detailed Description	128
3.34.2	Constructor & Destructor Documentation	128
3.34.2.1	TadiranCoral	128
3.34.2.2	TadiranCoral	128
3.35	TransferCallInfo Class Reference	129
3.35.1	Detailed Description	129
3.35.2	Property Documentation	129
3.35.2.1	ActiveDeviceCallID	129
3.35.2.2	ActiveDeviceNumber	129
3.35.2.3	ConnectedDevice	129
3.35.2.4	HeldDevice	129
3.35.2.5	HeldDeviceCallID	129
3.35.2.6	HeldDeviceNumber	129
3.36	UnifyOpenscapeVoice Class Reference	130
3.36.1	Detailed Description	130
3.36.2	Constructor & Destructor Documentation	130
3.36.2.1	UnifyOpenscapeVoice	130
3.36.2.2	UnifyOpenscapeVoice	130
3.37	UnifyOpenscapeX5 Class Reference	131
3.37.1	Detailed Description	131
3.37.2	Constructor & Destructor Documentation	131
3.37.2.1	UnifyOpenscapeX5	131
3.37.2.2	UnifyOpenscapeX5	131
3.38	VodiaSNOMOne Class Reference	132
3.38.1	Detailed Description	132
3.38.2	Constructor & Destructor Documentation	132

3.38.2.1	VodiaSNOMOne	132
3.38.2.2	VodiaSNOMOne	132
3.38.3	Member Function Documentation	132
3.38.3.1	MonitorStart	132
3.38.3.2	StartSession	133
3.38.3.3	StartSession	133

Chapter 1

CSTADLL

The CSTADLL product is a Microsoft .NET 4.5 DLL that allows client code to use a single library to communicate with a PBX device, regardless of what CSTA phase that device uses.

The DLL uses the following namespaces:

- `Com.Objsys.Csta.Common`
- `Com.Objsys.Csta.Devices`
- `Com.Objsys.Csta.Phase1`
- `Com.Objsys.Csta.Phase2`
- `Com.Objsys.Csta.Phase3`
- `Com.Objsys.Csta.Xml`

The `Com.Objsys.Csta.Common` namespace contains classes that are common to all CSTA phases.

The `Com.Objsys.Csta.Devices` namespace contains classes that allow a caller to use specific PBX devices.

The `Com.Objsys.Csta.Phase(n)` namespaces contain classes that are specific to the indicated phase. Most of these classes are generated by ASN1C from the CSTA and ACSE ASN.1 specifications. These generated classes are not documented here, but you can consult the ASN1C C# User Guide for information about how ASN.1 constructions are translated into C# classes.

The `Com.Objsys.Csta.Xml` namespace contains classes that are specific to using XML CSTA (edition 3, or ed3). Most of these classes are generated by XBinder from the CSTA and session management (ECMA-354) XML schema specifications. These generated classes are not documented here, but you can consult the XBinder Java/C# User Guide for information about how XML schema constructions are translated into C# classes.

Each namespace also contains several classes that are not generated by ASN1C. These classes are the ones documented in this manual.

A typical way to use the DLL is to use the `PBXSession` class to set up the communication to the PBX system via the constructor. If the PBX will be sending asynchronous data, such as monitor packets, to the client, the `ClientCallback` or `XMLClientCallback` property can be used to define a callback method to receive the asynchronous data. If no callback method is defined, asynchronous data will be ignored.

The CSTADLL kit includes some samples to guide you in writing your own code. The samples are evenly split between those implemented in C# and those implemented in Visual BASIC. Each language has samples for communicating with PBX devices that use BER CSTA and with PBX devices that use XML CSTA.

The classes and methods exposed by the all-phases DLL within the CSTADLL package are probably sufficient to handle operations for most PBX devices. But if needed, you can write a class of your own to handle operations for a PBX device that the CSTADLL software doesn't explicitly support. The sample NewPBX shows how this might be accomplished. This sample contains code for a small separate DLL that could be used to support a fictitious PBX device. The assumption in the sample is that this device uses standard messages for all operations except for the initial association messages. These messages are the ones that are most commonly different from one PBX to the next. The NewPBX sample shows how the `EncodeACSEConnectionRequest()` method within the `GenericCSTAp2` class (for BER PBX devices) or the `EncodeStartSession()` method within the `GenericXML` class (for XML PBX devices) can be overridden in a class that you can write. The override implementation handles the details that are specific to the device.

The all-phases DLL can log message traffic between a client program and the PBX device if so desired. The logging is controlled by the `LoggingEnabled` property with the `PBXSessionHelper` class. The logging is off by default. Both of the provided sample clients enable the logging. The log file used is named `cstadll_<program>.log`, where `<program>` is the name of the executable image that is using the DLL. The location of the log file is the folder where the executable image resides. If the log file grows to more than 5 Mb, it is copied to `cstadll_<program>.backup.log`, and a new log file is opened. If there is already a file with the backup file name, it is overwritten.

Chapter 2

Namespace Documentation

2.1 Package `Com.Objsys.Csta.Common`

Classes

- class [CSTAContext](#)
- class [CSTAResponseInfo](#)
- class [LicenseException](#)
- class [PBXSession](#)
- class [PBXSessionException](#)
- class [PBXSessionHelper](#)
- class [ResetSessionInfo](#)
- class [ROSEParseInfo](#)
- class [SocketState](#)

2.1.1 Detailed Description

The namespace `Com.Objsys.Csta.Common` contains classes that are common to all phases.

2.2 Package Com.Objsys.Csta.Devices

Classes

- class [Alcatel4400](#)
- class [AlcatelOXE](#)
- class [AlcatelOXO](#)
- class [PanasonicKXTDA](#)
- class [PanasonicKXTDE](#)
- class [PanasonicNCP](#)
- class [PhilipsSopho](#)
- class [SiemensCap](#)
- class [SiemensHicom300](#)
- class [SiemensHipath3000p2](#)
- class [SiemensHipath3000p3](#)
- class [SiemensHipath4000](#)
- class [SiemensRealitis](#)
- class [TadiranCoral](#)
- class [UnifyOpenscapeVoice](#)
- class [UnifyOpenscapeX5](#)
- class [VodiaSNOMOne](#)

2.2.1 Detailed Description

The namespace `Com.Objsys.Csta.Devices` contains classes that allow a caller to use specific PBX devices. The caller does not need to know what CSTA phase a device uses unless the device can accept messages formatted according to rules from more than one CSTA phase. In that case the class name ends with 'p(n)', where (n) is the number of the phase.

2.3 Package Com.Objsys.Csta.Phase1

Classes

- class [GenericCSTAp1](#)
- class [IETF_CSTAp1](#)
- class [Phase1OpCodes](#)

2.3.1 Detailed Description

The namespace [Com.Objsys.Csta.Phase1](#) contains classes that are specific to phase 1. Most of these classes are generated by ASN1C from the CSTA and ACSE ASN.1 specifications. These generated classes are not documented here, but you can consult the ASN1C C# User Guide for information about how ASN.1 constructions are translated into C# classes.

The namespace also contains several classes that are not generated by ASN1C. These classes are the ones documented in this manual.

2.4 Package Com.Objsys.Csta.Phase2

Classes

- class [GenericCSTAp2](#)
- class [IETF_CSTAp2](#)
- class [Phase2Opcodes](#)

2.4.1 Detailed Description

The namespace [Com.Objsys.Csta.Phase2](#) contains classes that are specific to phase 2. Most of these classes are generated by ASN1C from the CSTA and ACSE ASN.1 specifications. These generated classes are not documented here, but you can consult the ASN1C C# User Guide for information about how ASN.1 constructions are translated into C# classes.

The namespace also contains several classes that are not generated by ASN1C. These classes are the ones documented in this manual.

2.5 Package Com.Objsys.Csta.Phase3

Classes

- class [GenericCSTAp3](#)
- class [IETF_CSTAp3](#)
- class [Phase3Opcodes](#)
- class [SingleStepTransferInfo](#)
- class [TransferCallInfo](#)

2.5.1 Detailed Description

The namespace [Com.Objsys.Csta.Phase3](#) contains classes that are specific to phase 3. Most of these classes are generated by ASN1C from the CSTA and ACSE ASN.1 specifications. These generated classes are not documented here, but you can consult the ASN1C C# User Guide for information about how ASN.1 constructions are translated into C# classes.

The namespace also contains several classes that are not generated by ASN1C. These classes are the ones documented in this manual.

2.6 Package Com.Objsys.Csta.Xml

Classes

- class [GenericXML](#)

2.6.1 Detailed Description

The namespace `Com.Objsys.Csta.Xml` contains classes that are specific to XML CSTA (edition 3, or ed3). Most of these classes are generated by XBinder from the CSTA and session management (ECMA-354) XML schema specifications. These generated classes are not documented here, but you can consult the XBinder Java/C# User Guide for information about how XML schema constructions are translated into C# classes.

The namespace also contains several classes that are not generated by XBinder. These classes are the ones documented in this manual.

Chapter 3

Class Documentation

3.1 Alcatel4400 Class Reference

Inherits [Com::Objsys::Csta::Phase2::IETF_CSTAp2](#).

Inherited by [AlcatelOXE](#).

Public Member Functions

- [Alcatel4400](#) ([PBXSession](#) sessionObject)
- [Alcatel4400](#) (string pbxSystem, int port)
- override [CSTAResponseInfo](#) [MakeACSEAssociation](#) ()

Protected Member Functions

- override int [EncodeACSEConnectionRequest](#) ([CSTAResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer)
- override int [EncodeMakeCallRequest](#) ([CSTAResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, string callingDevice, string calledDevice)

3.1.1 Detailed Description

Implements CSTA phase 2 operations for the Alcatel 4400 PBX device.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 [Alcatel4400](#) (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

pbxSystem Well-known name or IP address of the PBX.

port Port on which the PBX listens for CSTA messages.

3.1.2.2 Alcatel4400 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.1.3 Member Function Documentation

3.1.3.1 override int EncodeACSEConnectionRequest (CSTARResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*) [protected, virtual]

Encodes an ACSE Association Request message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented from [GenericCSTAp2](#).

3.1.3.2 override int EncodeMakeCallRequest (CSTARResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, string *callingDevice*, string *calledDevice*) [protected, virtual]

Encodes a MakeCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callingDevice Identifier (e.g., phone number) of the device making the call.

calledDevice Identifier (e.g., phone number) of the device being called.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented from [GenericCSTAp2](#).

3.1.3.3 override CSTARResponseInfo MakeACSEAssociation () [virtual]

Establish an ACSE association with the PBX.

Returns

A CSTARResponseInfo object.

Reimplemented from [GenericCSTAp2](#).

3.2 AlcatelOXE Class Reference

Inherits [Com::Objsys::Csta::Devices::Alcatel4400](#).

Public Member Functions

- [AlcatelOXE](#) ([PBXSession](#) sessionObject)
- [AlcatelOXE](#) (string pbxSystem, int port)

3.2.1 Detailed Description

Implements CSTA phase 2 operations for the Alcatel OXE PBX device.

3.2.2 Constructor & Destructor Documentation

3.2.2.1 [AlcatelOXE](#) (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
port Port on which the PBX listens for CSTA messages.

3.2.2.2 [AlcatelOXE](#) ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

- sessionObject* A [PBXSession](#) object.

3.3 AlcatelOXO Class Reference

Inherits [Com::Objsys::Csta::Phase1::IETF_CSTAp1](#).

Public Member Functions

- [AlcatelOXO](#) ([PBXSession](#) sessionObject)
- [AlcatelOXO](#) (string pbxSystem, int port)
- override [CSTAResponseInfo](#) [MakeACSEAssociation](#) ()

Protected Member Functions

- override int [EncodeACSEConnectionRequest](#) ([CSTAResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encode-Buffer)

3.3.1 Detailed Description

Implements CSTA phase 1 operations for the Alcatel OXO device.

3.3.2 Constructor & Destructor Documentation

3.3.2.1 [AlcatelOXO](#) (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

pbxSystem Well-known name or IP address of the PBX.

port Port on which the PBX listens for CSTA messages.

3.3.2.2 [AlcatelOXO](#) ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

sessionObject A [PBXSession](#) object.

3.3.3 Member Function Documentation

3.3.3.1 override int [EncodeACSEConnectionRequest](#) ([CSTAResponseInfo](#) *response*, [Asn1BerEncodeBuffer](#) *encodeBuffer*) [**protected**, **virtual**]

Encodes an ACSE Association Request message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented from [GenericCSTAp1](#).

3.3.3.2 override CSTAResponseInfo MakeACSEAssociation () [virtual]

Establish an ACSE association with the PBX.

Returns

A CSTAResponseInfo object.

Reimplemented from [GenericCSTAp1](#).

3.4 CSTAContext Class Reference

Properties

- byte[] [ResponseFromPBX](#) [get, set]
- List< byte[] > [ResponsesFromPBX](#) [get, set]
- string [XMLResponseFromPBX](#) [get, set]
- List< string > [XMLResponsesFromPBX](#) [get, set]

3.4.1 Detailed Description

The [CSTAContext](#) class contains information needed to manage the interaction between the thread and the PBX.

3.4.2 Property Documentation

3.4.2.1 byte [] ResponseFromPBX [get, set]

See documentation for [CSTARResponseInfo.ResponseFromPBX](#).

3.4.2.2 List<byte[]> ResponsesFromPBX [get, set]

See documentation for [CSTARResponseInfo.ResponsesFromPBX](#).

3.4.2.3 string XMLResponseFromPBX [get, set]

See documentation for [CSTARResponseInfo.XMLResponseFromPBX](#).

3.4.2.4 List<string> XMLResponsesFromPBX [get, set]

See documentation for [CSTARResponseInfo.XMLResponsesFromPBX](#).

3.5 CSTAResponseInfo Class Reference

Properties

- byte[] [ResponseFromPBX](#) [get, set]
- List< byte[] > [ResponsesFromPBX](#) [get, set]
- int [StatusCode](#) [get, set]
- string [StatusMessage](#) [get, set]
- string [XMLResponseFromPBX](#) [get, set]
- List< string > [XMLResponsesFromPBX](#) [get, set]

3.5.1 Detailed Description

Contains information about a PBX operation that was attempted.

3.5.2 Property Documentation

3.5.2.1 byte [] ResponseFromPBX [get, set]

Contains the response from the PBX for messages that generate a single atomic response, or the immediate acknowledgement response for messages that generate multiple data responses (e.g., Get Switching Function [Devices](#)). If a message that normally generates multiple response segments encounters an error (e.g., the PBX rejects the message), then the single error message returned by the PBX will be in this property; the ResponsesFromPBX property will be null.

For CSTA operations this property is simply a reference to the ResponseFromPBX property of the thread's [CSTAContext](#) object. If the value of that property changes, then the value of this property changes.

3.5.2.2 List<byte[]> ResponsesFromPBX [get, set]

Contains the responses from the PBX for messages that generate multiple response segments (e.g., Get Switching Function [Devices](#)). If such a message encounters an error (e.g., the PBX rejects the message), then the single error message returned by the PBX will be in the ResponseFromPBX property; this property will be null. In all cases the first response, which is the acknowledgement message from the PBX, will be in the ResponseFromPBX property.

This property is simply a reference to the ResponsesFromPBX property of the thread's [CSTAContext](#) object. If the value of that property changes, then the value of this property changes.

3.5.2.3 int StatusCode [get, set]

A numeric status code. A value less than zero indicates that something went wrong during the attempted operation.

3.5.2.4 string StatusMessage [get, set]

Text containing information about a PBX operation that has completed, either successfully or not.

3.5.2.5 string XMLResponseFromPBX [get, set]

Contains the response from the PBX for messages that generate a single atomic XML response, or the immediate XML acknowledgement response for messages that generate multiple XML data responses (e.g., GetSwitchingFunctionDe-

vices). If a message that normally generates multiple response segments encounters an error (e.g., the PBX rejects the message), then the single error message returned by the PBX will be in this property; the ResponsesFromPBX property will be null.

For CSTA operations this property is simply a reference to the XMLResponseFromPBX property of the thread's [CSTAContext](#) object. If the value of that property changes, then the value of this property changes.

3.5.2.6 List<string> XMLResponsesFromPBX [get, set]

Contains the responses from the PBX for messages that generate multiple XML response segments (e.g., GetSwitchingFunctionDevices). If such a message encounters an error (e.g., the PBX rejects the message), then the single error message returned by the PBX will be in the ResponseFromPBX property; this property will be null. In all cases the first response, which is the acknowledgement message from the PBX, will be in the ResponseFromPBX property.

This property is simply a reference to the XMLResponsesFromPBX property of the thread's [CSTAContext](#) object. If the value of that property changes, then the value of this property changes.

3.6 GenericCSTAp1 Class Reference

Inherited by [SiemensHicom300](#), and [IETF_CSTAp1](#).

Public Member Functions

- virtual [CSTARResponseInfo AnswerCall](#) (ConnectionID callToAnswer, string deviceToLift)
- virtual [CSTARResponseInfo AnswerCall](#) (ConnectionID callToAnswer)
- virtual [CSTARResponseInfo AnswerCall](#) (string deviceToLift)
- virtual [CSTARResponseInfo ClearConnection](#) (ConnectionID connToClear)
- virtual [CSTARResponseInfo ClearDoNotDisturb](#) (string targetDevice)
- virtual [CSTARResponseInfo ClearMessageWaiting](#) (string targetDevice)
- virtual [CSTARResponseInfo ConsultationCall](#) (ConnectionID existingCall, string targetDevice)
- virtual [CSTARResponseInfo DivertCall](#) (string divertFrom, string divertTo)
- int [EncodeROSERequestHeader](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, [Phase1Opcodes.Opcodes](#) opcode)
- [GenericCSTAp1](#) ([PBXSession](#) sessionObject)
- [GenericCSTAp1](#) (string pbxSystem, int port)
- virtual [CSTARResponseInfo MakeACSEAssociation](#) ()
- virtual [CSTARResponseInfo MakeCall](#) (string callingDevice, string calledDevice)
- virtual [CSTARResponseInfo MonitorStart](#) (ConnectionID callToMonitor)
- virtual [CSTARResponseInfo MonitorStart](#) (string deviceToMonitor)
- virtual [CSTARResponseInfo MonitorStop](#) (string monitoredDevice)
- virtual [CSTARResponseInfo MonitorStop](#) (MonitorCrossRefID crossRefID)
- virtual [CSTARResponseInfo MonitorStop](#) (int crossRefInt)
- virtual [CSTARResponseInfo QueryDevice](#) (string deviceToQuery)
- virtual [CSTARResponseInfo ReleaseACSEAssociation](#) ()
- virtual [CSTARResponseInfo RetrieveCall](#) (ConnectionID callToRetrieve)
- virtual [CSTARResponseInfo SetDoNotDisturb](#) (string targetDevice)
- virtual [CSTARResponseInfo SetMessageWaiting](#) (string targetDevice)
- virtual [CSTARResponseInfo SnapshotDevice](#) (string deviceToSnapshot)
- virtual [CSTARResponseInfo TransferCall](#) (string heldDevice, string connectedDevice)
- virtual [CSTARResponseInfo TransferCall](#) (ConnectionID initiatedCall, ConnectionID originalCall)

Protected Member Functions

- virtual int [EncodeACSEConnectionRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer)
- virtual int [EncodeACSEReleaseRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer)
- virtual int [EncodeAnswerCallRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, ConnectionID callToAnswer, string deviceToLift)
- virtual int [EncodeAnswerCallRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, ConnectionID callToAnswer)
- virtual int [EncodeAnswerCallRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, string deviceToLift)
- virtual int [EncodeClearConnectionRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, ConnectionID connToClear)
- virtual int [EncodeConsultationCallRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, ConnectionID existingCall, string targetDevice)

- virtual int [EncodeDivertCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string divertFrom, string divertTo)
- virtual int [EncodeMakeCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string callingDevice, string calledDevice)
- virtual int [EncodeMonitorStartRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToMonitor)
- virtual int [EncodeMonitorStartRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToMonitor)
- virtual int [EncodeMonitorStopRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, MonitorCrossRefID crossRefObj)
- virtual int [EncodeMonitorStopRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, int crossRefInt, out MonitorCrossRefID crossRefID)
- virtual int [EncodeQueryDeviceRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToQuery)
- virtual int [EncodeRetrieveCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToRetrieve)
- virtual int [EncodeSetOrClearDNDRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, bool dndOn)
- virtual int [EncodeSetOrClearMWRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, bool indicatorOn)
- virtual int [EncodeSnapshotDeviceRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToSnapshot)
- virtual int [EncodeTransferCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string heldDevice, string connectedDevice)
- virtual int [EncodeTransferCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID initiatedCall, ConnectionID originalCall)

Properties

- [PBXSession SessionObject](#) [get]
- [CSTAContext ThreadContext](#) [get]

3.6.1 Detailed Description

Implements CSTA phase 1 operations using BER. Note that most PBXes don't support all CSTA messages, so some methods in this class may result in an error status being returned by your PBX.

3.6.2 Constructor & Destructor Documentation

3.6.2.1 GenericCSTAp1 (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
- port* Port on which the PBX listens for CSTA messages.

3.6.2.2 GenericCSTAp1 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.6.3 Member Function Documentation

3.6.3.1 virtual CSTAResponseInfo AnswerCall (ConnectionID *callToAnswer*, string *deviceToLift*) [virtual]

Answers a call.

Parameters

callToAnswer The connection id of the call to answer.

deviceToLift The device (e.g., extension number) that is to answer the call.

Returns

A CSTAResponseInfo object.

3.6.3.2 virtual CSTAResponseInfo AnswerCall (ConnectionID *callToAnswer*) [virtual]

Answers a call.

Parameters

callToAnswer The ConnectionID of the call to answer.

Returns

A CSTAResponseInfo object.

3.6.3.3 virtual CSTAResponseInfo AnswerCall (string *deviceToLift*) [virtual]

Answers a call.

Parameters

deviceToLift The identification (e.g., phone number) of the device to answer.

Returns

A CSTAResponseInfo object.

3.6.3.4 virtual CSTAResponseInfo ClearConnection (ConnectionID *connToClear*) [virtual]

Clears a connection.

Parameters

connToClear The ConnectionID of the connection to clear.

Returns

A CSTAResponseInfo object.

3.6.3.5 virtual CSTAResponseInfo ClearDoNotDisturb (string *targetDevice*) [virtual]

Turns off the Do Not Disturb functionality for a phone.

Parameters

targetDevice The device for which the Do Not Disturb functionality is to be turned off.

Returns

A CSTAResponseInfo object.

3.6.3.6 virtual CSTAResponseInfo ClearMessageWaiting (string *targetDevice*) [virtual]

Turns off the message waiting indicator on a device's display.

Parameters

targetDevice The device for which the indicator is to be turned off.

Returns

A CSTAResponseInfo object.

3.6.3.7 virtual CSTAResponseInfo ConsultationCall (ConnectionID *existingCall*, string *targetDevice*) [virtual]

Instruct the PBX to do a consultation call.

Parameters

existingCall The connection id of the call for which the consultation call will be made.

targetDevice Identifier (e.g., phone number) of the device that is the target of the consultation call.

Returns

A CSTAResponseInfo object.

3.6.3.8 virtual CSTAResponseInfo DivertCall (string *divertFrom*, string *divertTo*) [virtual]

Diverts a call from a source to a destination.

Parameters

divertFrom Identifier (e.g., phone number) of the call to be diverted.

divertTo Identifier (e.g., phone number) of the location to which the call is to be diverted.

Returns

A CSTAResponseInfo object.

3.6.3.9 virtual int EncodeACSEConnectionRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer) [protected, virtual]

Encodes an ACSE Association Request message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented in [AlcatelOXO](#).

3.6.3.10 virtual int EncodeACSEReleaseRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer) [protected, virtual]

Encodes an ACSE Release Request message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.11 virtual int EncodeAnswerCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToAnswer, string deviceToLift) [protected, virtual]

Encodes an AnswerCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToAnswer The ConnectionID of the call to answer.

deviceToLift The identification (e.g., phone number) of the device to answer.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.12 virtual int EncodeAnswerCallRequest (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, ConnectionID *callToAnswer*) [protected, virtual]

Encodes an AnswerCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToAnswer The ConnectionID of the call to answer.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.13 virtual int EncodeAnswerCallRequest (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, string *deviceToLift*) [protected, virtual]

Encodes an AnswerCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

deviceToLift The identification (e.g., phone number) of the device to answer.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.14 virtual int EncodeClearConnectionRequest (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, ConnectionID *connToClear*) [protected, virtual]

Encodes a ClearConnection message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

connToClear The ConnectionID of the connection to clear.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.15 virtual int EncodeConsultationCallRequest (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, ConnectionID *existingCall*, string *targetDevice*) [protected, virtual]

Encodes a ConsultationCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

existingCall The connection id of the call for which the consultation call will be made.

targetDevice Identifier (e.g., phone number) of the device that is the target of the consultation call.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.16 virtual int EncodeDivertCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string divertFrom, string divertTo) [protected, virtual]

Encodes a DivertCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

divertFrom Identifier (e.g., phone number) of the device from which the call is to be diverted.

divertTo Identifier (e.g., phone number) of the device to which the call is to be diverted.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.17 virtual int EncodeMakeCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string callingDevice, string calledDevice) [protected, virtual]

Encodes a MakeCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callingDevice Identifier (e.g., phone number) of the device making the call.

calledDevice Identifier (e.g., phone number) of the device being called.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.18 virtual int EncodeMonitorStartRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToMonitor) [protected, virtual]

Encodes a MonitorStart message to monitor a call.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToMonitor The call to monitor.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.19 `virtual int EncodeMonitorStartRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToMonitor) [protected, virtual]`

Encodes a MonitorStart message to monitor a device.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

deviceToMonitor Identifier (e.g., telephone number) of the device to monitor.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.20 `virtual int EncodeMonitorStopRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, MonitorCrossRefID crossRefObj) [protected, virtual]`

Encodes a MonitorStop message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

crossRefObj The cross reference id of the monitor request as a MonitorCrossRefID object.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.21 `virtual int EncodeMonitorStopRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, int crossRefInt, out MonitorCrossRefID crossRefID) [protected, virtual]`

Encodes a MonitorStop message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

crossRefInt The cross reference id of the monitor request as an integer.

crossRefID A reference to a variable of type MonitorCrossRefID that will be populated with a MonitorCrossRefID object that corresponds to the specified integer cross reference id.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.22 virtual int EncodeQueryDeviceRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToQuery) [protected, virtual]

Encodes a QueryDevice message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

deviceToQuery The identification (e.g., phone number) of the device to query.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.23 virtual int EncodeRetrieveCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToRetrieve) [protected, virtual]

Encodes a RetrieveCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToRetrieve The ConnectionID of the call to retrieve.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.24 int EncodeROSERequestHeader (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, Phase1Opcodes.Opcodes opcode)

This method prepends a ROSE header to an already encoded phase 1 CSTA message.

Parameters

response A CSTAResponseInfo object, used to communicate any exception information back to the caller.

encodeBuffer An Asn1BerEncodeBuffer instance containing the already encoded CSTA message.

opcode The opcode enumeration for the operation that the encoded CSTA message describes.

Returns

The length of the encoded message, including both the CSTA message and the ROSE header, or -1 if the encoding fails.

3.6.3.25 virtual int EncodeSetOrClearDNDRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, bool dndOn) [protected, virtual]

Encodes a SetFuture message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice The device for which Do Not Disturb is to be set or cleared.

dndOn If true, DoNotDisturb will be turned on. If false, DoNotDisturb will be turned off.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.26 `virtual int EncodeSetOrClearMWRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, bool indicatorOn) [protected, virtual]`

Encodes a SetFeature message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice The device for which the message waiting indicator is to be turned on or off.

indicatorOn If true, the message waiting indicator is turned on. If false, the message waiting indicator is turned off.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.27 `virtual int EncodeSnapshotDeviceRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToSnapshot) [protected, virtual]`

Encodes a SnapshotDevice message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

deviceToSnapshot Identifier (e.g., phone number) of the device for which the snapshot is desired.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.28 `virtual int EncodeTransferCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string heldDevice, string connectedDevice) [protected, virtual]`

Encodes a TransferCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

heldDevice Identifier (e.g., phone number) of the device from which the call is transferred.

connectedDevice Identifier (e.g., phone number) of the device to which the call is transferred.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.29 `virtual int EncodeTransferCallRequest (CSTARResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID initiatedCall, ConnectionID originalCall) [protected, virtual]`

Encodes a TransferCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

initiatedCall ConnectionID of the new call initiated by the consultation call. The initiatedCall member of the ConsultationCallResult class, for example, contains this ConnectionID.

originalCall ConnectionID of the original call. The somewhat confusingly named callingDevice member of the MakeCallResult class contains this ConnectionID, as does the establishedConnection member of the EstablishedEvent class.

Returns

The length of the encoded message, or -1 if an error occurred.

3.6.3.30 `virtual CSTARResponseInfo MakeACSEAssociation () [virtual]`

Establish an ACSE association with the PBX.

Returns

A CSTARResponseInfo object.

Reimplemented in [AlcatelOXO](#), and [PhilipsSopho](#).

3.6.3.31 `virtual CSTARResponseInfo MakeCall (string callingDevice, string calledDevice) [virtual]`

Instruct the PBX to place a call.

Parameters

callingDevice Identifier (e.g., phone number) of the device making the call.

calledDevice Identifier (e.g., phone number) of the device being called.

Returns

A CSTARResponseInfo object.

3.6.3.32 virtual CSTAResponseInfo MonitorStart (ConnectionID *callToMonitor*) [virtual]

Issues a MonitorStart request to the PBX to monitor a call.

Parameters

callToMonitor The call to monitor.

Returns

A CSTAResponseInfo object.

3.6.3.33 virtual CSTAResponseInfo MonitorStart (string *deviceToMonitor*) [virtual]

Issues a MonitorStart request to the PBX to monitor a device.

Parameters

deviceToMonitor Identifier (e.g., telephone number) of the device to monitor.

Returns

A CSTAResponseInfo object.

3.6.3.34 virtual CSTAResponseInfo MonitorStop (string *monitoredDevice*) [virtual]

This method stops all monitors active against the indicated device, regardless of what thread started the monitor. The method will only stop monitors started through the [MonitorStart\(\)](#) method.

Parameters

monitoredDevice The monitored device (e.g., extension).

Returns

If no problems are encountered, the method returns a CSTAResponseInfo object containing the response from the PBX for the LAST Monitor Stop message.

If any problems are encountered, the method returns a CSTAResponseInfo object containing information about the error, including any response from the PBX for the problematic Monitor Stop message.

3.6.3.35 virtual CSTAResponseInfo MonitorStop (MonitorCrossRefID *crossRefID*) [virtual]

Stop a previously started PBX monitor request.

Parameters

crossRefID The cross reference id of the monitor request as a MonitorCrossRefID object.

Returns

A CSTAResponseInfo object.

3.6.3.36 virtual CSTAResponseInfo MonitorStop (int *crossRefInt*) [virtual]

Stop a previously started PBX monitor request.

Parameters

crossRefInt The cross reference id of the monitor request as an integer.

Returns

A CSTAResponseInfo object.

3.6.3.37 virtual CSTAResponseInfo QueryDevice (string *deviceToQuery*) [virtual]

Queries a device.

Parameters

deviceToQuery The identification (e.g., phone number) of the device to query.

Returns

A CSTAResponseInfo object.

3.6.3.38 virtual CSTAResponseInfo ReleaseACSEAssociation () [virtual]

Releases an ACSE association with a PBX device.

Returns

A CSTAResponseInfo object. For this message the connection with the PBX is closed, so null is returned.

3.6.3.39 virtual CSTAResponseInfo RetrieveCall (ConnectionID *callToRetrieve*) [virtual]

Retrieves a held call.

Parameters

callToRetrieve The ConnectionID of the call to retrieve.

Returns

A CSTAResponseInfo object.

3.6.3.40 virtual CSTAResponseInfo SetDoNotDisturb (string *targetDevice*) [virtual]

Sets the Do Not Disturb feature for a phone.

Parameters

targetDevice The device for which Do Not Disturb is to be set.

Returns

A CSTAResponseInfo object.

3.6.3.41 virtual CSTAResponseInfo SetMessageWaiting (string *targetDevice*) [virtual]

Turns on the message waiting indicator on a device's display.

Parameters

targetDevice The device for which the indicator is to be turned on.

Returns

A CSTAResponseInfo object.

3.6.3.42 virtual CSTAResponseInfo SnapshotDevice (string *deviceToSnapshot*) [virtual]

Instruct the PBX to take a snapshot of calls active at a device.

Parameters

deviceToSnapshot Identifier (e.g., phone number) of the device for which the snapshot is desired.

Returns

A CSTAResponseInfo object.

3.6.3.43 virtual CSTAResponseInfo TransferCall (string *heldDevice*, string *connectedDevice*) [virtual]

Transfers a call from one device to another.

Parameters

heldDevice Identifier (e.g., phone number) of the device from which the call is transferred.

connectedDevice Identifier (e.g., phone number) of the device to which the call is transferred.

Returns

A CSTAResponseInfo object.

3.6.3.44 virtual CSTAResponseInfo TransferCall (ConnectionID *initiatedCall*, ConnectionID *originalCall*) [virtual]

Transfers a call. A consultation call must be done before calling this method.

Parameters

initiatedCall ConnectionID of the new call initiated by the consultation call. The *initiatedCall* member of the ConsultationCallResult class, for example, contains this ConnectionID.

originalCall ConnectionID of the original call. The somewhat confusingly named *callingDevice* member of the MakeCallResult class contains this ConnectionID, as does the *establishedConnection* member of the EstablishedEvent class.

Returns

A CSTAResponseInfo object.

3.6.4 Property Documentation

3.6.4.1 PBXSession SessionObject [get]

The PBXSession object associated with this instance.

3.6.4.2 CSTAContext ThreadContext [get]

The CSTAContext structure for this thread.

3.7 GenericCSTAp2 Class Reference

Inherited by [SiemensHipath3000p2](#), and [IETF_CSTAp2](#).

Public Member Functions

- virtual [CSTARResponseInfo AnswerCall](#) (ConnectionID callToAnswer, string deviceToLift)
- virtual [CSTARResponseInfo AnswerCall](#) (ConnectionID callToAnswer)
- virtual [CSTARResponseInfo AnswerCall](#) (string deviceToLift)
- virtual [CSTARResponseInfo ClearConnection](#) (ConnectionID connToClear)
- virtual [CSTARResponseInfo ClearDoNotDisturb](#) (string targetDevice)
- virtual [CSTARResponseInfo ClearMessageWaiting](#) (string targetDevice)
- virtual [CSTARResponseInfo ConsultationCall](#) (ConnectionID existingCall, string targetDevice)
- virtual [CSTARResponseInfo DivertCall](#) (string divertFrom, string divertTo)
- int [EncodeROSERequestHeader](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, [Phase2Opcodes.Opcodes](#) opcode)
- [GenericCSTAp2](#) ([PBXSession](#) sessionObject)
- [GenericCSTAp2](#) (string pbxSystem, int port)
- virtual [CSTARResponseInfo MakeACSEAssociation](#) ()
- virtual [CSTARResponseInfo MakeCall](#) (string callingDevice, string calledDevice)
- virtual [CSTARResponseInfo MonitorStart](#) (ConnectionID callToMonitor)
- virtual [CSTARResponseInfo MonitorStart](#) (string deviceToMonitor)
- virtual [CSTARResponseInfo MonitorStop](#) (string monitoredDevice)
- virtual [CSTARResponseInfo MonitorStop](#) (MonitorCrossRefID crossRefID)
- virtual [CSTARResponseInfo MonitorStop](#) (int crossRefInt)
- virtual [CSTARResponseInfo QueryDevice](#) (string deviceToQuery)
- virtual [CSTARResponseInfo ReleaseACSEAssociation](#) ()
- virtual [CSTARResponseInfo RetrieveCall](#) (ConnectionID callToRetrieve)
- virtual [CSTARResponseInfo SetDoNotDisturb](#) (string targetDevice)
- virtual [CSTARResponseInfo SetMessageWaiting](#) (string targetDevice)
- virtual [CSTARResponseInfo SnapshotDevice](#) (string deviceToSnapshot)
- virtual [CSTARResponseInfo TransferCall](#) (string heldDevice, string connectedDevice)
- virtual [CSTARResponseInfo TransferCall](#) (ConnectionID initiatedCall, ConnectionID originalCall)

Protected Member Functions

- virtual int [EncodeACSEConnectionRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer)
- virtual int [EncodeACSEReleaseRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer)
- virtual int [EncodeAnswerCallRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, ConnectionID callToAnswer, string deviceToLift)
- virtual int [EncodeAnswerCallRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, ConnectionID callToAnswer)
- virtual int [EncodeAnswerCallRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, string deviceToLift)
- virtual int [EncodeClearConnectionRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, ConnectionID connToClear)
- virtual int [EncodeConsultationCallRequest](#) ([CSTARResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, ConnectionID existingCall, string targetDevice)

- virtual int [EncodeDivertCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string divertFrom, string divertTo)
- virtual int [EncodeMakeCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string callingDevice, string calledDevice)
- virtual int [EncodeMonitorStartRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToMonitor)
- virtual int [EncodeMonitorStartRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToMonitor)
- virtual int [EncodeMonitorStopRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, MonitorCrossRefID crossRefObj)
- virtual int [EncodeMonitorStopRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, int crossRefInt, out MonitorCrossRefID crossRefID)
- virtual int [EncodeQueryDeviceRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToQuery)
- virtual int [EncodeRetrieveCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToRetrieve)
- virtual int [EncodeSetOrClearDNDRquest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, bool dndOn)
- virtual int [EncodeSetOrClearMWRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, bool indicatorOn)
- virtual int [EncodeSnapshotDeviceRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToSnapshot)
- virtual int [EncodeTransferCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string heldDevice, string connectedDevice)
- virtual int [EncodeTransferCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID initiatedCall, ConnectionID originalCall)

Properties

- [PBXSession SessionObject](#) [get]
- [CSTAContext ThreadContext](#) [get]

3.7.1 Detailed Description

Implements CSTA phase 2 operations using BER. Note that most PBXes don't support all CSTA messages, so some methods in this class may result in an error status being returned by your PBX.

3.7.2 Constructor & Destructor Documentation

3.7.2.1 GenericCSTAp2 (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
- port* Port on which the PBX listens for CSTA messages.

3.7.2.2 GenericCSTAp2 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.7.3 Member Function Documentation

3.7.3.1 virtual CSTAResponseInfo AnswerCall (ConnectionID *callToAnswer*, string *deviceToLift*) [virtual]

Answers a call.

Parameters

callToAnswer The connection id of the call to answer.

deviceToLift The device (e.g., extension number) that is to answer the call.

Returns

A CSTAResponseInfo object.

3.7.3.2 virtual CSTAResponseInfo AnswerCall (ConnectionID *callToAnswer*) [virtual]

Answers a call.

Parameters

callToAnswer The ConnectionID of the call to answer.

Returns

A CSTAResponseInfo object.

3.7.3.3 virtual CSTAResponseInfo AnswerCall (string *deviceToLift*) [virtual]

Answers a call.

Parameters

deviceToLift The identification (e.g., phone number) of the device to answer.

Returns

A CSTAResponseInfo object.

3.7.3.4 virtual CSTAResponseInfo ClearConnection (ConnectionID *connToClear*) [virtual]

Clears a connection.

Parameters

connToClear The ConnectionID of the connection to clear.

Returns

A CSTAResponseInfo object.

3.7.3.5 virtual CSTAResponseInfo ClearDoNotDisturb (string *targetDevice*) [virtual]

Turns off the Do Not Disturb functionality for a phone.

Parameters

targetDevice The device for which the Do Not Disturb functionality is to be turned off.

Returns

A CSTAResponseInfo object.

3.7.3.6 virtual CSTAResponseInfo ClearMessageWaiting (string *targetDevice*) [virtual]

Turns off the message waiting indicator on a device's display.

Parameters

targetDevice The device for which the indicator is to be turned off.

Returns

A CSTAResponseInfo object.

3.7.3.7 virtual CSTAResponseInfo ConsultationCall (ConnectionID *existingCall*, string *targetDevice*) [virtual]

Instruct the PBX to do a consultation call.

Parameters

existingCall The connection id of the call for which the consultation call will be made.

targetDevice Identifier (e.g., phone number) of the device that is the target of the consultation call.

Returns

A CSTAResponseInfo object.

3.7.3.8 virtual CSTAResponseInfo DivertCall (string *divertFrom*, string *divertTo*) [virtual]

Diverts a call from a source to a destination.

Parameters

divertFrom Identifier (e.g., phone number) of the device from which the call is to be diverted.

divertTo Identifier (e.g., phone number) of the device to which the call is to be diverted.

Returns

A CSTAResponseInfo object.

3.7.3.9 virtual int EncodeACSEConnectionRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer) [protected, virtual]

Encodes an ACSE Association Request message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented in [Alcatel4400](#).

3.7.3.10 virtual int EncodeACSEReleaseRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer) [protected, virtual]

Encodes an ACSE Release Request message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.11 virtual int EncodeAnswerCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToAnswer, string deviceToLift) [protected, virtual]

Encodes an AnswerCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToAnswer The ConnectionID of the call to answer.

deviceToLift The identification (e.g., phone number) of the device to answer.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.12 virtual int EncodeAnswerCallRequest (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, ConnectionID *callToAnswer*) [protected, virtual]

Encodes an AnswerCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToAnswer The ConnectionID of the call to answer.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.13 virtual int EncodeAnswerCallRequest (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, string *deviceToLift*) [protected, virtual]

Encodes an AnswerCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

deviceToLift The identification (e.g., phone number) of the device to answer.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.14 virtual int EncodeClearConnectionRequest (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, ConnectionID *connToClear*) [protected, virtual]

Encodes a ClearConnection message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

connToClear The ConnectionID of the connection to clear.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.15 virtual int EncodeConsultationCallRequest (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, ConnectionID *existingCall*, string *targetDevice*) [protected, virtual]

Encodes a ConsultationCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

existingCall The connection id of the call for which the consultation call will be made.

targetDevice Identifier (e.g., phone number) of the device that is the target of the consultation call.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.16 `virtual int EncodeDivertCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string divertFrom, string divertTo) [protected, virtual]`

Encodes a DivertCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

divertFrom Identifier (e.g., phone number) of the device from which the call is to be diverted.

divertTo Identifier (e.g., phone number) of the device to which the call is to be diverted.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.17 `virtual int EncodeMakeCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string callingDevice, string calledDevice) [protected, virtual]`

Encodes a MakeCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callingDevice Identifier (e.g., phone number) of the device making the call.

calledDevice Identifier (e.g., phone number) of the device being called.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented in [Alcate14400](#).

3.7.3.18 `virtual int EncodeMonitorStartRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToMonitor) [protected, virtual]`

Encodes a MonitorStart message to monitor a call.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToMonitor The call to monitor.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.19 **virtual int EncodeMonitorStartRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToMonitor) [protected, virtual]**

Encodes a MonitorStart message to monitor a device.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

deviceToMonitor Identifier (e.g., telephone number) of the device to monitor.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.20 **virtual int EncodeMonitorStopRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, MonitorCrossRefID crossRefObj) [protected, virtual]**

Encodes a MonitorStop message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

crossRefObj The cross reference id of the monitor request as a MonitorCrossRefID object.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.21 **virtual int EncodeMonitorStopRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, int crossRefInt, out MonitorCrossRefID crossRefID) [protected, virtual]**

Encodes a MonitorStop message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

crossRefInt The cross reference id of the monitor request as an integer.

crossRefID A reference to a variable of type MonitorCrossRefID that will be populated with a MonitorCrossRefID object that corresponds to the specified integer cross reference id.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.22 `virtual int EncodeQueryDeviceRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToQuery) [protected, virtual]`

Encodes a QueryDevice message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

deviceToQuery The identification (e.g., phone number) of the device to query.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.23 `virtual int EncodeRetrieveCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToRetrieve) [protected, virtual]`

Encodes a RetrieveCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToRetrieve The ConnectionID of the call to retrieve.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.24 `int EncodeROSERequestHeader (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, Phase2Opcodes.Opcodes opcode)`

This method prepends a ROSE header to an already encoded phase 2 CSTA message.

Parameters

response A CSTAResponseInfo object, used to communicate any exception information back to the caller.

encodeBuffer An Asn1BerEncodeBuffer instance containing the already encoded CSTA message.

opcode The opcode enumeration for the operation that the encoded CSTA message describes.

Returns

The length of the encoded message, including both the CSTA message and the ROSE header, or -1 if the encoding fails.

3.7.3.25 virtual int EncodeSetOrClearDNDRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, bool dndOn) [protected, virtual]

Encodes a SetFeature message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice The device for which Do Not Disturb is to be set or cleared.

dndOn If true, DoNotDisturb will be turned on. If false, DoNotDisturb will be turned off.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.26 virtual int EncodeSetOrClearMWRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, bool indicatorOn) [protected, virtual]

Encodes a SetFeature message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice The device for which the indicator is to be turned on or off.

indicatorOn If true, the message waiting indicator is turned on. If false, the message waiting indicator is turned off.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.27 virtual int EncodeSnapshotDeviceRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToSnapshot) [protected, virtual]

Encodes a SnapshotDevice message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

deviceToSnapshot Identifier (e.g., phone number) of the device for which the snapshot is desired.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.28 `virtual int EncodeTransferCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string heldDevice, string connectedDevice) [protected, virtual]`

Encodes a TransferCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

heldDevice Identifier (e.g., phone number) of the device from which the call is transferred.

connectedDevice Identifier (e.g., phone number) of the device to which the call is transferred.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.29 `virtual int EncodeTransferCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID initiatedCall, ConnectionID originalCall) [protected, virtual]`

Encodes a TransferCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

initiatedCall ConnectionID of the new call initiated by the consultation call. The initiatedCall member of the ConsultationCallResult class, for example, contains this ConnectionID.

originalCall ConnectionID of the original call. The somewhat confusingly named callingDevice member of the MakeCallResult class contains this ConnectionID, as does the establishedConnection member of the EstablishedEvent class.

Returns

The length of the encoded message, or -1 if an error occurred.

3.7.3.30 `virtual CSTAResponseInfo MakeACSEAssociation () [virtual]`

Establish an ACSE association with the PBX.

Returns

A CSTAResponseInfo object.

Reimplemented in [Alcatel4400](#).

3.7.3.31 `virtual CSTAResponseInfo MakeCall (string callingDevice, string calledDevice) [virtual]`

Instruct the PBX to place a call.

Parameters

callingDevice Identifier (e.g., phone number) of the device making the call.

calledDevice Identifier (e.g., phone number) of the device being called.

Returns

A CSTAResponseInfo object.

3.7.3.32 virtual CSTAResponseInfo MonitorStart (ConnectionID *callToMonitor*) [virtual]

Issues a MonitorStart request to the PBX to monitor a call.

Parameters

callToMonitor The call to monitor.

Returns

A CSTAResponseInfo object.

3.7.3.33 virtual CSTAResponseInfo MonitorStart (string *deviceToMonitor*) [virtual]

Issues a MonitorStart request to the PBX to monitor a device.

Parameters

deviceToMonitor Identifier (e.g., telephone number) of the device to monitor.

Returns

A CSTAResponseInfo object.

3.7.3.34 virtual CSTAResponseInfo MonitorStop (string *monitoredDevice*) [virtual]

This method stops all monitors active against the indicated device, regardless of what thread started the monitor. The method will only stop monitors started through the [MonitorStart\(\)](#) method.

Parameters

monitoredDevice The monitored device (e.g., extension).

Returns

If no problems are encountered, the method returns a CSTAResponseInfo object containing the response from the PBX for the LAST Monitor Stop message.

If any problems are encountered, the method returns a CSTAResponseInfo object containing information about the error, including any response from the PBX for the problematic Monitor Stop message.

3.7.3.35 virtual CSTAResponseInfo MonitorStop (MonitorCrossRefID *crossRefID*) [virtual]

Stop a previously started PBX monitor request.

Parameters

crossRefID The cross reference id of the monitor request as a MonitorCrossRefID object.

Returns

A CSTAResponseInfo object.

3.7.3.36 virtual CSTAResponseInfo MonitorStop (int *crossRefInt*) [virtual]

Stop a previously started PBX monitor request.

Parameters

crossRefInt The cross reference id of the monitor request as an integer.

Returns

A CSTAResponseInfo object.

3.7.3.37 virtual CSTAResponseInfo QueryDevice (string *deviceToQuery*) [virtual]

Queries a device.

Parameters

deviceToQuery The identification (e.g., phone number) of the device to query.

Returns

A CSTAResponseInfo object.

3.7.3.38 virtual CSTAResponseInfo ReleaseACSEAssociation () [virtual]

Releases an ACSE association with a PBX device.

Returns

A CSTAResponseInfo object. For this message the connection with the PBX is closed, so null is returned.

3.7.3.39 virtual CSTAResponseInfo RetrieveCall (ConnectionID *callToRetrieve*) [virtual]

Retrieves a held call.

Parameters

callToRetrieve The ConnectionID of the call to retrieve.

Returns

A CSTAResponseInfo object.

3.7.3.40 virtual CSTAResponseInfo SetDoNotDisturb (string *targetDevice*) [virtual]

Sets the Do Not Disturb feature for a phone.

Parameters

targetDevice The device for which Do Not Disturb is to be set.

Returns

A CSTAResponseInfo object.

3.7.3.41 virtual CSTAResponseInfo SetMessageWaiting (string *targetDevice*) [virtual]

Turns on the message waiting indicator on a device's display.

Parameters

targetDevice The device for which the indicator is to be turned on.

Returns

A CSTAResponseInfo object.

3.7.3.42 virtual CSTAResponseInfo SnapshotDevice (string *deviceToSnapshot*) [virtual]

Instruct the PBX to take a snapshot of calls active at a device.

Parameters

deviceToSnapshot Identifier (e.g., phone number) of the device for which the snapshot is desired.

Returns

A CSTAResponseInfo object.

3.7.3.43 virtual CSTAResponseInfo TransferCall (string *heldDevice*, string *connectedDevice*) [virtual]

Transfers a call from one device to another.

Parameters

heldDevice Identifier (e.g., phone number) of the device from which the call is transferred.

connectedDevice Identifier (e.g., phone number) of the device to which the call is transferred.

Returns

A CSTAResponseInfo object.

3.7.3.44 virtual CSTAResponseInfo TransferCall (ConnectionID *initiatedCall*, ConnectionID *originalCall*) [virtual]

Transfers a call. A consultation call must be done before calling this method.

Parameters

initiatedCall ConnectionID of the new call initiated by the consultation call. The *initiatedCall* member of the *ConsultationCallResult* class, for example, contains this ConnectionID.

originalCall ConnectionID of the original call. The somewhat confusingly named *callingDevice* member of the *MakeCallResult* class contains this ConnectionID, as does the *establishedConnection* member of the *EstablishedEvent* class.

Returns

A *CSTAResponseInfo* object.

3.7.4 Property Documentation

3.7.4.1 PBXSession SessionObject [get]

The *PBXSession* object associated with this instance.

3.7.4.2 CSTAContext ThreadContext [get]

The *CSTAContext* structure for this thread.

3.8 GenericCSTAp3 Class Reference

Inherited by [SiemensHipath3000p3](#), and [IETF_CSTAp3](#).

Public Member Functions

- virtual [CSTARResponseInfo AlternateCall](#) (ConnectionID heldCall, ConnectionID activeCall)
- virtual [CSTARResponseInfo AnswerCall](#) (ConnectionID callToAnswer, string deviceToLift)
- virtual [CSTARResponseInfo AnswerCall](#) (ConnectionID callToAnswer)
- virtual [CSTARResponseInfo AnswerCall](#) (string deviceToLift)
- virtual [CSTARResponseInfo ClearConnection](#) (ConnectionID connToClear)
- virtual [CSTARResponseInfo ClearDoNotDisturb](#) (string targetDevice)
- virtual [CSTARResponseInfo ClearMessageWaiting](#) (string targetDevice)
- virtual [CSTARResponseInfo ConferenceCall](#) (ConnectionID heldCall, ConnectionID activeCall)
- virtual [CSTARResponseInfo ConsultationCall](#) (ConnectionID existingCall, string targetDevice)
- virtual int [EncodeROSERequestHeader](#) ([CSTARResponseInfo](#) response, Asn1BerEncodeBuffer encodeBuffer, Phase3Opcodes.Opcodes opcode)
- [GenericCSTAp3](#) ([PBXSession](#) sessionObject)
- [GenericCSTAp3](#) (string pbxSystem, int port)
- virtual [CSTARResponseInfo GetAgentState](#) (string agentDevice)
- virtual [CSTARResponseInfo GetDoNotDisturb](#) (string targetDevice)
- virtual [CSTARResponseInfo GetLogicalDevInfo](#) (string targetDevice)
- virtual [CSTARResponseInfo GetPhysicalDevInfo](#) (string targetDevice)
- virtual [CSTARResponseInfo GetSFDevices](#) (ReqDeviceCategory deviceCategory)
- virtual [CSTARResponseInfo GetSFDevices](#) ()
- virtual [CSTARResponseInfo HoldCall](#) (ConnectionID callToHold)
- virtual [CSTARResponseInfo MakeACSEAssociation](#) ()
- virtual [CSTARResponseInfo MakeCall](#) (string callingDevice, string calledDevice)
- virtual [CSTARResponseInfo MonitorStart](#) (ConnectionID callToMonitor)
- virtual [CSTARResponseInfo MonitorStart](#) (string deviceToMonitor)
- virtual [CSTARResponseInfo MonitorStop](#) (string monitoredDevice)
- virtual [CSTARResponseInfo MonitorStop](#) (MonitorCrossRefID crossRefID)
- virtual [CSTARResponseInfo MonitorStop](#) (int crossRefInt)
- virtual [CSTARResponseInfo ReleaseACSEAssociation](#) ()
- virtual [CSTARResponseInfo RetrieveCall](#) (ConnectionID callToRetrieve)
- virtual [CSTARResponseInfo RingDevice](#) (string targetDevice, string targetRinger, long ringPattern)
- virtual [CSTARResponseInfo SendData](#) (IOCrossRefID ioCrossRef, string text)
- virtual [CSTARResponseInfo SetAgentState](#) (string agentDevice, ReqAgentState agentState)
- virtual [CSTARResponseInfo SetAgentState](#) (string agentDevice, ReqAgentState agentState, AgentID aid)
- virtual [CSTARResponseInfo SetDisplay](#) (string targetDevice, string text)
- virtual [CSTARResponseInfo SetDoNotDisturb](#) (string targetDevice)
- virtual [CSTARResponseInfo SetMessageWaiting](#) (string targetDevice)
- virtual [CSTARResponseInfo SingleStepTransfer](#) ([SingleStepTransferInfo](#) sstInfo)
- virtual [CSTARResponseInfo SingleStepTransfer](#) (ConnectionID callToTransfer, string transferToDevice)
- virtual [CSTARResponseInfo SnapshotCall](#) (ConnectionID callToSnapshot)
- virtual [CSTARResponseInfo SnapshotDevice](#) (string deviceToSnapshot)
- virtual [CSTARResponseInfo StartDataPath](#) (string targetDevice)
- virtual [CSTARResponseInfo StopDataPath](#) (IOCrossRefID ioCrossRef)
- virtual [CSTARResponseInfo StopRing](#) (string targetDevice, string targetRinger, long ringPattern)
- virtual [CSTARResponseInfo TransferCall](#) ([TransferCallInfo](#) tcInfo)
- virtual [CSTARResponseInfo TransferCall](#) (ConnectionID initiatedCall, ConnectionID originalCall)

Protected Member Functions

- virtual int [EncodeACSEConnectionRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer)
- virtual int [EncodeACSEReleaseRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer)
- virtual int [EncodeAlternateCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID heldCall, ConnectionID activeCall)
- virtual int [EncodeAnswerCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToAnswer, string deviceToLift)
- virtual int [EncodeAnswerCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToAnswer)
- virtual int [EncodeAnswerCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToLift)
- virtual int [EncodeClearConnectionRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID connToClear)
- virtual int [EncodeConferenceCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID heldCall, ConnectionID activeCall)
- virtual int [EncodeConsultationCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID existingCall, string targetDevice)
- virtual int [EncodeGetAgentStateRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string agentDevice)
- virtual int [EncodeGetDNDRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice)
- virtual int [EncodeGetLogicalDevInfoRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice)
- virtual int [EncodeGetPhysicalDevInfoRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice)
- virtual int [EncodeGetSFDevicesRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ReqDeviceCategory deviceCategory)
- virtual int [EncodeHoldCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToHold)
- virtual int [EncodeMakeCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string callingDevice, string calledDevice)
- virtual int [EncodeMonitorStartRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToMonitor)
- virtual int [EncodeMonitorStartRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToMonitor)
- virtual int [EncodeMonitorStopRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, MonitorCrossRefID crossRefObj)
- virtual int [EncodeMonitorStopRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, int crossRefInt, out MonitorCrossRefID crossRefID)
- virtual int [EncodeRetrieveCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToRetrieve)
- virtual int [EncodeSendDataRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, IOCrossRefID ioCrossRef, string text)
- virtual int [EncodeSetAgentStateRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string agentDevice, ReqAgentState agentState)
- virtual int [EncodeSetAgentStateRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string agentDevice, ReqAgentState agentState, AgentID aid)
- virtual int [EncodeSetDisplayRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, string text)
- virtual int [EncodeSetOrClearDNDRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, bool dndOn)

- virtual int [EncodeSetOrClearMWRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, bool indicatorOn)
- virtual int [EncodeSetRingerStatusRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, string targetRinger, RingMode mode, long ringPattern)
- virtual int [EncodeSingleStepTransferRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, SingleStepTransferInfo sstInfo)
- virtual int [EncodeSingleStepTransferRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToTransfer, string transferToDevice)
- virtual int [EncodeSnapshotCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToSnapshot)
- virtual int [EncodeSnapshotDeviceRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToSnapshot)
- virtual int [EncodeStartDataPathRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice)
- virtual int [EncodeStopDataPathRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, IOCrossRefID ioCrossRef)
- virtual int [EncodeTransferCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, TransferCallInfo tcInfo)
- virtual int [EncodeTransferCallRequest](#) (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID initiatedCall, ConnectionID originalCall)

Properties

- [PBXSession SessionObject](#) [get]
- [CSTAContext ThreadContext](#) [get]

3.8.1 Detailed Description

Implements CSTA phase 3 operations using BER. Note that most PBXes don't support all CSTA messages, so some methods in this class may result in an error status being returned by your PBX.

3.8.2 Constructor & Destructor Documentation

3.8.2.1 GenericCSTAp3 (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
- port* Port on which the PBX listens for CSTA messages.

3.8.2.2 GenericCSTAp3 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

- sessionObject* A PBXSession object.

3.8.3 Member Function Documentation

3.8.3.1 virtual CSTAResponseInfo AlternateCall (ConnectionID *heldCall*, ConnectionID *activeCall*) [virtual]

Places an existing active call on hold and then either retrieves a previously held call or connects to an alerting or queued call at the same device (e.g., to answer a call waiting call).

Parameters

heldCall The held call to be retrieved.

activeCall The active call to be placed on hold.

Returns

A CSTAResponseInfo object.

3.8.3.2 virtual CSTAResponseInfo AnswerCall (ConnectionID *callToAnswer*, string *deviceToLift*) [virtual]

Answers a call.

Parameters

callToAnswer ConnectionID of an existing call (such as initiated through [MakeCall\(\)](#)).

deviceToLift The device (e.g., "800") that is to answer the call.

Returns

A CSTAResponseInfo object.

3.8.3.3 virtual CSTAResponseInfo AnswerCall (ConnectionID *callToAnswer*) [virtual]

Answers a call.

Parameters

callToAnswer The ConnectionID of the call to answer.

Returns

A CSTAResponseInfo object.

3.8.3.4 virtual CSTAResponseInfo AnswerCall (string *deviceToLift*) [virtual]

Answers a call.

Parameters

deviceToLift The identification (e.g., phone number) of the device to answer.

Returns

A CSTAResponseInfo object.

3.8.3.5 virtual CSTAResponseInfo ClearConnection (ConnectionID *connToClear*) [virtual]

Clears a connection.

Parameters

connToClear The ConnectionID of the connection to clear.

Returns

A CSTAResponseInfo object.

3.8.3.6 virtual CSTAResponseInfo ClearDoNotDisturb (string *targetDevice*) [virtual]

Turns off the Do Not Disturb functionality for a phone.

Parameters

targetDevice The device for which the Do Not Disturb functionality is to be turned off.

Returns

A CSTAResponseInfo object.

3.8.3.7 virtual CSTAResponseInfo ClearMessageWaiting (string *targetDevice*) [virtual]

Turns off the message waiting indicator on a device's display.

Parameters

targetDevice The device for which the indicator is to be turned off.

Returns

A CSTAResponseInfo object.

3.8.3.8 virtual CSTAResponseInfo ConferenceCall (ConnectionID *heldCall*, ConnectionID *activeCall*) [virtual]

Brings a held call into conference with an active call.

Parameters

heldCall The held call to be brought into conference.

activeCall The active call.

Returns

A CSTAResponseInfo object.

3.8.3.9 virtual CSTAResponseInfo ConsultationCall (ConnectionID *existingCall*, string *targetDevice*) [virtual]

Instruct the PBX to do a consultation call.

Parameters

existingCall The connection id of the call for which the consultation call will be made.

targetDevice Identifier (e.g., phone number) of the device that is the target of the consultation call.

Returns

A CSTAResponseInfo object.

3.8.3.10 virtual int EncodeACSEConnectionRequest (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*) [protected, virtual]

Encodes an ACSE Association Request message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented in [PanasonicKXTDE](#), [SiemensCap](#), and [SiemensHipath3000p3](#).

3.8.3.11 virtual int EncodeACSEReleaseRequest (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*) [protected, virtual]

Encodes an ACSE Release Request message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.12 virtual int EncodeAlternateCallRequest (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, ConnectionID *heldCall*, ConnectionID *activeCall*) [protected, virtual]

Encodes an AlternateCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

heldCall ConnectionID of the held call to be retrieved.

activeCall ConnectionID of the active call to be placed on hold.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.13 **virtual int EncodeAnswerCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToAnswer, string deviceToLift) [protected, virtual]**

Encodes an AnswerCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToAnswer The ConnectionID of the call to answer.

deviceToLift The identification (e.g., phone number) of the device to answer.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.14 **virtual int EncodeAnswerCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToAnswer) [protected, virtual]**

Encodes an AnswerCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToAnswer The ConnectionID of the call to answer.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.15 **virtual int EncodeAnswerCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToLift) [protected, virtual]**

Encodes an AnswerCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

deviceToLift The identification (e.g., phone number) of the device to answer.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.16 `virtual int EncodeClearConnectionRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID connToClear) [protected, virtual]`

Encodes a ClearConnection message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

connToClear The ConnectionID of the connection to clear.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.17 `virtual int EncodeConferenceCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID heldCall, ConnectionID activeCall) [protected, virtual]`

Encodes a ConferenceCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

heldCall ConnectionID of the held call to be retrieved.

activeCall ConnectionID of the active call.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.18 `virtual int EncodeConsultationCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID existingCall, string targetDevice) [protected, virtual]`

Encodes a ConsultationCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

existingCall The connection id of the call for which the consultation call will be made.

targetDevice Identifier (e.g., phone number) of the device that is the target of the consultation call.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.19 `virtual int EncodeGetAgentStateRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string agentDevice) [protected, virtual]`

Encodes a GetAgentState message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

agentDevice The device associated with the agent.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.20 `virtual int EncodeGetDNDRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice) [protected, virtual]`

Encodes a GetDoNotDisturb message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice The phone for which the Do Not Disturb setting is desired.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.21 `virtual int EncodeGetLogicalDevInfoRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice) [protected, virtual]`

Encodes a GetLogicalDeviceInformation message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice The device for which the information is needed.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.22 `virtual int EncodeGetPhysicalDevInfoRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice) [protected, virtual]`

Encodes a GetPhysicalDeviceInformation message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice The device for which the information is needed.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.23 `virtual int EncodeGetSFDevicesRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ReqDeviceCategory deviceCategory) [protected, virtual]`

Encodes a GetSwitchingFunctionDevices message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

deviceCategory The category of device for which the list is desired.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.24 `virtual int EncodeHoldCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToHold) [protected, virtual]`

Encodes a HoldCall message.

Parameters

response A CSTAResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToHold The ConnectionID of the call to be held.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.25 `virtual int EncodeMakeCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string callingDevice, string calledDevice) [protected, virtual]`

Encodes a MakeCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callingDevice Identifier (e.g., phone number) of the device making the call.

calledDevice Identifier (e.g., phone number) of the device being called.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented in [PanasonicKXTDE](#).

3.8.3.26 `virtual int EncodeMonitorStartRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToMonitor) [protected, virtual]`

Encodes a MonitorStart message to monitor a call.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToMonitor The call to monitor.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.27 `virtual int EncodeMonitorStartRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToMonitor) [protected, virtual]`

Encodes a MonitorStart message to monitor a device.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

deviceToMonitor Identifier (e.g., telephone number) of the device to monitor.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.28 `virtual int EncodeMonitorStopRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, MonitorCrossRefID crossRefObj) [protected, virtual]`

Encodes a MonitorStop message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

crossRefObj The cross reference id of the monitor request as a MonitorCrossRefID object.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.29 virtual int EncodeMonitorStopRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, int crossRefInt, out MonitorCrossRefID crossRefID) [protected, virtual]

Encodes a MonitorStop message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

crossRefInt The cross reference id of the monitor request as an integer.

crossRefID A reference to a variable of type MonitorCrossRefID that will be populated with a MonitorCrossRefID object that corresponds to the specified integer cross reference id.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.30 virtual int EncodeRetrieveCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToRetrieve) [protected, virtual]

Encodes a RetrieveCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToRetrieve The ConnectionID of the call to retrieve.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.31 virtual int EncodeROSERequestHeader (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, Phase3OpCodes.OpCodes opcode) [virtual]

This method prepends a ROSE header to an already encoded phase 3 CSTA message.

Parameters

response A CSTAResponseInfo object, used to communicate any exception information back to the caller.

encodeBuffer An Asn1BerEncodeBuffer instance containing the already encoded CSTA message.

opcode The opcode enumeration for the operation that the encoded CSTA message describes.

Returns

The length of the encoded message, including both the CSTA message and the ROSE header, or -1 if the encoding fails.

3.8.3.32 virtual int EncodeSendDataRequest (CSTARResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, IOCrossRefID *ioCrossRef*, string *text*) [protected, virtual]

Encodes a SendData message.

Parameters

response A CSTARResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

ioCrossRef An IOCrossRefID object, most likely obtained by a previous call to StartDataPath.

text The text to send to the telephony device.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.33 virtual int EncodeSetAgentStateRequest (CSTARResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, string *agentDevice*, ReqAgentState *agentState*) [protected, virtual]

Encodes a SetAgentState message.

Parameters

response A CSTARResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

agentDevice The device associated with the agent.

agentState An ReqAgentState object indicating the desired state of the agent.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.34 virtual int EncodeSetAgentStateRequest (CSTARResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, string *agentDevice*, ReqAgentState *agentState*, AgentID *aid*) [protected, virtual]

Encodes a SetAgentState message.

Parameters

response A CSTARResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

agentDevice The device associated with the agent.

agentState An ReqAgentState object indicating the desired state of the agent.

aid The agent id as an AgentID object.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.35 virtual int EncodeSetDisplayRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, string text) [protected, virtual]

Encodes a SetDisplay message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice The device to which the text is to be sent.

text The text to be sent.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented in [PanasonicKXTDE](#).

3.8.3.36 virtual int EncodeSetOrClearDNDRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, bool dndOn) [protected, virtual]

Encodes a SetDoNotDisturb message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice The device for which Do Not Disturb is to be set or cleared.

dndOn If true, DoNotDisturb will be turned on. If false, DoNotDisturb will be turned off.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.37 virtual int EncodeSetOrClearMWRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, bool indicatorOn) [protected, virtual]

Encodes a SetMessageWaiting message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice The device for which the indicator is to be turned on or off.

indicatorOn If true, the message waiting indicator is turned on. If false, the message waiting indicator is turned off.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.38 `virtual int EncodeSetRingerStatusRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, string targetRinger, RingMode mode, long ringPattern) [protected, virtual]`

Encodes a SetRingerStatus message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice The device whose ringer is to be affected.

targetRinger The id of the ringer to use for the operation. This argument can be specified as a character string (e.g, "abc"), a hex string (e.g, "'010A05'H"), or a binary string (e.g, "'000000010000101000000101'B").

mode A RingMode object that specifies ringing or notRinging.

ringPattern The indicator of the ring pattern to use.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.39 `virtual int EncodeSingleStepTransferRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, SingleStepTransferInfo sstInfo) [protected, virtual]`

Encodes a SingleStepTransfer message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

sstInfo A [SingleStepTransferInfo](#) object.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented in [SiemensHipath3000p3](#).

3.8.3.40 `virtual int EncodeSingleStepTransferRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToTransfer, string transferToDevice) [protected, virtual]`

Encodes a SingleStepTransfer message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToTransfer The ConnectionID of the call to transfer.

transferToDevice The device to which the call is to be transferred.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.41 virtual int EncodeSnapshotCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID callToSnapshot) [protected, virtual]

Encodes a SnapshotCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callToSnapshot The ConnectionID of the call for which the snapshot is desired.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.42 virtual int EncodeSnapshotDeviceRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string deviceToSnapshot) [protected, virtual]

Encodes a SnapshotDevice message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

deviceToSnapshot Identifier (e.g., phone number) of the device for which the snapshot is desired.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.43 virtual int EncodeStartDataPathRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice) [protected, virtual]

Encodes a StartDataPath message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice Specifies the device to which a data path is to be opened.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.44 virtual int EncodeStopDataPathRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, IOCrossRefID ioCrossRef) [protected, virtual]

Encodes a StopDataPath message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

ioCrossRef An IOCrossRef object, most likely obtained from a previous call to StartDataPath.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.45 `virtual int EncodeTransferCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, TransferCallInfo tcInfo) [protected, virtual]`

Encodes a TransferCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

tcInfo A [TransferCallInfo](#) object.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.46 `virtual int EncodeTransferCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, ConnectionID initiatedCall, ConnectionID originalCall) [protected, virtual]`

Encodes a TransferCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

initiatedCall ConnectionID of the new call initiated by the consultation call. The *initiatedCall* member of the ConsultationCallResult class, for example, contains this ConnectionID.

originalCall ConnectionID of the original call. The somewhat confusingly named *callingDevice* member of the MakeCallResult class contains this ConnectionID, as does the *establishedConnection* member of the EstablishedEvent class.

Returns

The length of the encoded message, or -1 if an error occurred.

3.8.3.47 `virtual CSTAResponseInfo GetAgentState (string agentDevice) [virtual]`

Gets the state of the agent associated with a device.

Parameters

agentDevice The device associated with the agent.

Returns

A CSTAResponseInfo object.

3.8.3.48 virtual CSTAResponseInfo GetDoNotDisturb (string *targetDevice*) [virtual]

Gets the Do Not Disturb setting for a phone.

Parameters

targetDevice The phone for which the Do Not Disturb setting is desired.

Returns

A CSTAResponseInfo object.

3.8.3.49 virtual CSTAResponseInfo GetLogicalDevInfo (string *targetDevice*) [virtual]

Gets information about the logical element of a device.

Parameters

targetDevice The device for which the information is desired.

Returns

A CSTAResponseInfo object.

3.8.3.50 virtual CSTAResponseInfo GetPhysicalDevInfo (string *targetDevice*) [virtual]

Gets information about the physical element of a device.

Parameters

targetDevice The device for which the information is desired.

Returns

A CSTAResponseInfo object.

3.8.3.51 virtual CSTAResponseInfo GetSFDevices (ReqDeviceCategory *deviceCategory*) [virtual]

Sends a Get Switching Function [Devices](#) request to the PBX.

Parameters

deviceCategory The category of device for which the list is desired.

Returns

A CSTAResponseInfo object.

3.8.3.52 virtual CSTAResponseInfo GetSFDevices () [virtual]

Sends a Get Switching Function [Devices](#) request to the PBX.

Returns

A CSTAResponseInfo object.

Reimplemented in [PanasonicKXTDE](#).

3.8.3.53 virtual CSTAResponseInfo HoldCall (ConnectionID *callToHold*) [virtual]

Instruct the PBX to hold a call.

Parameters

callToHold The ConnectionID of the call to be held.

Returns

A CSTAResponseInfo object.

3.8.3.54 virtual CSTAResponseInfo MakeACSEAssociation () [virtual]

Establish an ACSE association with the PBX.

Returns

A CSTAResponseInfo object.

3.8.3.55 virtual CSTAResponseInfo MakeCall (string *callingDevice*, string *calledDevice*) [virtual]

Instruct the PBX to place a call.

Parameters

callingDevice Identifier (e.g., phone number) of the device making the call.

calledDevice Identifier (e.g., phone number) of the device being called.

Returns

A CSTAResponseInfo object.

3.8.3.56 virtual CSTAResponseInfo MonitorStart (ConnectionID *callToMonitor*) [virtual]

Issues a MonitorStart request to the PBX to monitor a call.

Parameters

callToMonitor The call to monitor.

Returns

A CSTAResponseInfo object.

3.8.3.57 virtual CSTAResponseInfo MonitorStart (string *deviceToMonitor*) [virtual]

Issues a MonitorStart request to the PBX to monitor a device.

Parameters

deviceToMonitor Identifier (e.g., telephone number) of the device to monitor.

Returns

A CSTAResponseInfo object.

3.8.3.58 virtual CSTAResponseInfo MonitorStop (string *monitoredDevice*) [virtual]

This method stops all monitors active against the indicated device, regardless of what thread started the monitor. The method will only stop monitors started through the [MonitorStart\(\)](#) method.

Parameters

monitoredDevice The monitored device (e.g., extension).

Returns

If no problems are encountered, the method returns a CSTAResponseInfo object containing the response from the PBX for the LAST Monitor Stop message.

If any problems are encountered, the method returns a CSTAResponseInfo object containing information about the error, including any response from the PBX for the problematic Monitor Stop message.

3.8.3.59 virtual CSTAResponseInfo MonitorStop (MonitorCrossRefID *crossRefID*) [virtual]

Stop a previously started PBX monitor request.

Parameters

crossRefID The cross reference id of the monitor request as a MonitorCrossRefID object.

Returns

A CSTAResponseInfo object.

3.8.3.60 virtual CSTAResponseInfo MonitorStop (int *crossRefInt*) [virtual]

Stop a previously started PBX monitor request.

Parameters

crossRefInt The cross reference id of the monitor request as an integer.

Returns

A CSTAResponseInfo object.

3.8.3.61 virtual CSTAResponseInfo ReleaseACSEAssociation () [virtual]

Releases an ACSE association with a PBX device.

Returns

A CSTAResponseInfo object. For this message the connection with the PBX is closed, so null is returned.

3.8.3.62 virtual CSTAResponseInfo RetrieveCall (ConnectionID *callToRetrieve*) [virtual]

Retrieves a held call.

Parameters

callToRetrieve The ConnectionID of the call to retrieve.

Returns

A CSTAResponseInfo object.

3.8.3.63 virtual CSTAResponseInfo RingDevice (string *targetDevice*, string *targetRinger*, long *ringPattern*) [virtual]

Causes a telephony device to ring.

Parameters

targetDevice The device to ring.

targetRinger The id of the ringer to use for the ring. This argument can be specified as a character string (e.g. "abc"), a hex string (e.g. "010A05'H"), or a binary string (e.g. "000000010000101000000101'B").

ringPattern The indicator of the ring pattern to use.

Returns

A CSTAResponseInfo object.

3.8.3.64 virtual CSTAResponseInfo SendData (IOCrossRefID *ioCrossRef*, string *text*) [virtual]

Sends a text message to a telephony device.

Parameters

ioCrossRef An IOCrossRefID object, most likely obtained by a previous call to StartDataPath.

text The text to send to the telephony device.

Returns

A CSTAResponseInfo object.

3.8.3.65 virtual CSTAResponseInfo SetAgentState (string *agentDevice*, ReqAgentState *agentState*) [virtual]

Sets the state of the agent associated with a device.

Parameters

agentDevice The device associated with the agent.

agentState The desired state for the agent.

Returns

A CSTAResponseInfo object.

3.8.3.66 virtual CSTAResponseInfo SetAgentState (string *agentDevice*, ReqAgentState *agentState*, AgentID *aid*) [virtual]

Sets the state of the agent associated with a device.

Parameters

agentDevice The device associated with the agent.

agentState The desired state for the agent.

aid The agent id.

Returns

A CSTAResponseInfo object.

3.8.3.67 virtual CSTAResponseInfo SetDisplay (string *targetDevice*, string *text*) [virtual]

Sends text to a telephony device's display

Parameters

targetDevice The device to which the text is to be sent.

text The text to be sent.

Returns

A CSTAResponseInfo object.

3.8.3.68 virtual CSTAResponseInfo SetDoNotDisturb (string *targetDevice*) [virtual]

Sets the Do Not Disturb feature for a phone.

Parameters

targetDevice The device for which Do Not Disturb is to be set.

Returns

A CSTAResponseInfo object.

3.8.3.69 virtual CSTAResponseInfo SetMessageWaiting (string *targetDevice*) [virtual]

Turns on the message waiting indicator on a device's display.

Parameters

targetDevice The device for which the indicator is to be turned on.

Returns

A CSTAResponseInfo object.

3.8.3.70 virtual CSTAResponseInfo SingleStepTransfer (SingleStepTransferInfo *sstInfo*) [virtual]

Perform a single step transfer.

Parameters

sstInfo A [SingleStepTransferInfo](#) object.

Returns

A CSTAResponseInfo object.

3.8.3.71 virtual CSTAResponseInfo SingleStepTransfer (ConnectionID *callToTransfer*, string *transferToDevice*) [virtual]

Perform a single step transfer.

Parameters

callToTransfer The ConnectionID of the call to transfer.

transferToDevice The device to which the call is to be transferred.

Returns

A CSTAResponseInfo object.

3.8.3.72 virtual CSTAResponseInfo SnapshotCall (ConnectionID *callToSnapshot*) [virtual]

Instruct the PBX to take a snapshot of a call.

Parameters

callToSnapshot The ConnectionID of the call for which the snapshot is desired.

Returns

A CSTAResponseInfo object.

3.8.3.73 virtual CSTAResponseInfo SnapshotDevice (string *deviceToSnapshot*) [virtual]

Instruct the PBX to take a snapshot of calls active at a device.

Parameters

deviceToSnapshot Identifier (e.g., phone number) of the device for which the snapshot is desired.

Returns

A CSTAResponseInfo object.

3.8.3.74 virtual CSTAResponseInfo StartDataPath (string *targetDevice*) [virtual]

Opens up a data path to a specified device.

Parameters

targetDevice Specifies the device to which a data path is to be opened.

Returns

A CSTAResponseInfo object.

3.8.3.75 virtual CSTAResponseInfo StopDataPath (IOCrossRefID *ioCrossRef*) [virtual]

Stops a previously established data path

Parameters

ioCrossRef An IOCrossRef object, most likely obtained from a previous call to StartDataPath.

Returns

A CSTAResponseInfo object.

3.8.3.76 virtual CSTAResponseInfo StopRing (string *targetDevice*, string *targetRinger*, long *ringPattern*) [virtual]

Stops a ringer on a telephony device.

Parameters

targetDevice The device for which the ringer is to stop.

targetRinger The id of the ringer to stop. This argument can be specified as a character string (e.g, "abc"), a hex string (e.g, "'010A05'H"), or a binary string (e.g, "'000000010000101000000101'B").

ringPattern The indicator of the ring pattern to stop.

Returns

A CSTAResponseInfo object.

3.8.3.77 virtual CSTAResponseInfo TransferCall (TransferCallInfo *tcInfo*) [virtual]

Transfers a call from one device to another.

Parameters

tcInfo A [TransferCallInfo](#) object.

Returns

A CSTAResponseInfo object.

3.8.3.78 virtual CSTAResponseInfo TransferCall (ConnectionID *initiatedCall*, ConnectionID *originalCall*) [virtual]

Transfers a call. A consultation call must be done before calling this method.

Parameters

initiatedCall ConnectionID of the new call initiated by the consultation call. The *initiatedCall* member of the [ConsultationCallResult](#) class, for example, contains this ConnectionID.

originalCall ConnectionID of the original call. The somewhat confusingly named *callingDevice* member of the [MakeCallResult](#) class contains this ConnectionID, as does the *establishedConnection* member of the [EstablishedEvent](#) class.

Returns

A CSTAResponseInfo object.

3.8.4 Property Documentation

3.8.4.1 PBXSession SessionObject [get]

The PBXSession object associated with this instance.

3.8.4.2 CSTAContext ThreadContext [get]

The CSTAContext structure for this thread.

3.9 GenericXML Class Reference

Inherited by [UnifyOpenscapeVoice](#), and [VodiaSNOMOne](#).

Public Member Functions

- virtual [CSTARResponseInfo AnswerCall](#) (ConnectionID callToAnswer, string deviceToLift)
- virtual [CSTARResponseInfo AnswerCall](#) (ConnectionID callToAnswer)
- virtual [CSTARResponseInfo ClearConnection](#) (ConnectionID connectionToClear)
- virtual [CSTARResponseInfo ClearDoNotDisturb](#) (string targetDevice)
- virtual [CSTARResponseInfo ClearMessageWaiting](#) (string targetDevice)
- virtual [CSTARResponseInfo ConsultationCall](#) (ConnectionID existingCall, string targetDevice)
- virtual string [EncodeInvokeID](#) (string initialMessage)
- [GenericXML](#) (PBXSession sessionObject)
- [GenericXML](#) (string pbxSystem, int port)
- virtual [CSTARResponseInfo GetAgentState](#) (string agentDevice)
- virtual [CSTARResponseInfo GetDoNotDisturb](#) (string targetDevice)
- virtual [CSTARResponseInfo GetSFDevices](#) ()
- virtual [CSTARResponseInfo GetSFDevices](#) (ReqDeviceCategory deviceCategory)
- virtual [CSTARResponseInfo HoldCall](#) (ConnectionID callToHold)
- virtual [CSTARResponseInfo MakeCall](#) (string callingDevice, string calledDevice)
- virtual [CSTARResponseInfo MonitorStart](#) (ConnectionID callToMonitor)
- virtual [CSTARResponseInfo MonitorStart](#) (string deviceToMonitor)
- virtual [CSTARResponseInfo MonitorStop](#) (string crossRefID)
- virtual [CSTARResponseInfo MonitorStopAtDevice](#) (string monitoredDevice)
- virtual [CSTARResponseInfo RetrieveCall](#) (ConnectionID callToRetrieve)
- virtual [CSTARResponseInfo RingDevice](#) (string targetDevice, string targetRinger, long ringPattern)
- virtual [CSTARResponseInfo SendData](#) (IOCrossRefID xref, string text)
- virtual [CSTARResponseInfo SetAgentState](#) (string agentDevice, ReqAgentState agentState)
- virtual [CSTARResponseInfo SetAgentState](#) (string agentDevice, ReqAgentState agentState, string agentID)
- virtual [CSTARResponseInfo SetDisplay](#) (string targetDevice, string text)
- virtual [CSTARResponseInfo SetDoNotDisturb](#) (string targetDevice)
- virtual [CSTARResponseInfo SetMessageWaiting](#) (string targetDevice)
- virtual [CSTARResponseInfo SingleStepTransfer](#) (ConnectionID callToTransfer, string transferToDevice)
- virtual [CSTARResponseInfo SnapshotCall](#) (ConnectionID callToSnapshot)
- virtual [CSTARResponseInfo SnapshotDevice](#) (string deviceToSnapshot)
- virtual [CSTARResponseInfo StartDataPath](#) (string targetDevice)
- virtual [CSTARResponseInfo StartSession](#) (string applicationID)
- virtual [CSTARResponseInfo StartSession](#) ()
- virtual [CSTARResponseInfo StopDataPath](#) (IOCrossRefID xref)
- virtual [CSTARResponseInfo StopRing](#) (string targetDevice, string targetRinger, long ringPattern)
- virtual [CSTARResponseInfo StopSession](#) ()
- virtual [CSTARResponseInfo TransferCall](#) (ConnectionID initiatedCall, ConnectionID originalCall)

Protected Member Functions

- virtual string [EncodeAnswerCall](#) (CSTARResponseInfo response, ConnectionID callToAnswer, string deviceToLift)
- virtual string [EncodeAnswerCall](#) (CSTARResponseInfo response, ConnectionID callToAnswer)
- virtual string [EncodeClearConnection](#) (CSTARResponseInfo response, ConnectionID connectionToClear)
- virtual string [EncodeConsultationCall](#) (CSTARResponseInfo response, ConnectionID existingCall, string targetDevice)
- virtual string [EncodeGetAgentState](#) (CSTARResponseInfo response, string targetDevice)
- virtual string [EncodeGetDND](#) (CSTARResponseInfo response, string targetDevice)
- virtual string [EncodeGetSFDevices](#) (CSTARResponseInfo response, ReqDeviceCategory category)
- virtual string [EncodeHoldCall](#) (CSTARResponseInfo response, ConnectionID callToHold)
- virtual string [EncodeMakeCall](#) (CSTARResponseInfo response, string callingDevice, string calledDevice)
- virtual string [EncodeMonitorStart](#) (CSTARResponseInfo response, ConnectionID targetCall)
- virtual string [EncodeMonitorStart](#) (CSTARResponseInfo response, string targetDevice)
- virtual string [EncodeMonitorStop](#) (CSTARResponseInfo response, string xref)
- virtual string [EncodeRetrieveCall](#) (CSTARResponseInfo response, ConnectionID callToRetrieve)
- virtual string [EncodeSendData](#) (CSTARResponseInfo response, IOCrossRefID xref, string strText)
- virtual string [EncodeSetAgentState](#) (CSTARResponseInfo response, string agentDevice, ReqAgentState agentState)
- virtual string [EncodeSetAgentState](#) (CSTARResponseInfo response, string agentDevice, ReqAgentState agentState, string agentID)
- virtual string [EncodeSetDisplay](#) (CSTARResponseInfo response, string targetDevice, string text)
- virtual string [EncodeSetDND](#) (CSTARResponseInfo response, string targetDevice, bool dndOn)
- virtual string [EncodeSetMsgWaiting](#) (CSTARResponseInfo response, string targetDevice, bool indicatorOn)
- virtual string [EncodeSetRingerStatus](#) (CSTARResponseInfo response, string targetDevice, string targetRinger, RingMode rm, long ringPattern)
- virtual string [EncodeSingleStepTransfer](#) (CSTARResponseInfo response, ConnectionID callToTransfer, string transferToDevice)
- virtual string [EncodeSnapshotCall](#) (CSTARResponseInfo response, ConnectionID callToSnapshot)
- virtual string [EncodeSnapshotDevice](#) (CSTARResponseInfo response, string targetDevice)
- virtual string [EncodeStartDataPath](#) (CSTARResponseInfo response, string targetDevice)
- virtual string [EncodeStartSession](#) (CSTARResponseInfo response, string applicationID)
- virtual string [EncodeStopDataPath](#) (CSTARResponseInfo response, IOCrossRefID xref)
- virtual string [EncodeStopSession](#) (CSTARResponseInfo response)
- virtual string [EncodeTransferCall](#) (CSTARResponseInfo response, ConnectionID initiatedCall, ConnectionID originalCall)

Properties

- [PBXSession SessionObject](#) [get]
- [CSTAContext ThreadContext](#) [get]

3.9.1 Detailed Description

Implements CSTA phase 3 operations using XML. Note that most PBXes don't support all CSTA messages, so some methods in this class may result in an error status being returned by your PBX.

3.9.2 Constructor & Destructor Documentation

3.9.2.1 GenericXML (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

pbxSystem Well-known name or IP address of the PBX.

port Port on which the PBX listens for CSTA messages.

3.9.2.2 GenericXML (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.9.3 Member Function Documentation

3.9.3.1 virtual CSTAResponseInfo AnswerCall (ConnectionID *callToAnswer*, string *deviceToLift*) [virtual]

Answers a call.

Parameters

callToAnswer ConnectionID of an existing call (such as initiated through [MakeCall\(\)](#)).

deviceToLift The device (e.g., "800") that is to answer the call.

Returns

A CSTAResponseInfo object.

3.9.3.2 virtual CSTAResponseInfo AnswerCall (ConnectionID *callToAnswer*) [virtual]

Answers a call.

Parameters

callToAnswer The ConnectionID of the call to answer.

Returns

A CSTAResponseInfo object.

3.9.3.3 virtual CSTAResponseInfo ClearConnection (ConnectionID *connectionToClear*) [virtual]

Clears a connection.

Parameters

connectionToClear The ConnectionID of the connection to clear.

Returns

A CSTAResponseInfo object.

3.9.3.4 virtual CSTAResponseInfo ClearDoNotDisturb (string *targetDevice*) [virtual]

Turns off the Do Not Disturb functionality for a phone.

Parameters

targetDevice The device for which the Do Not Disturb functionality is to be turned off.

Returns

A CSTAResponseInfo object.

3.9.3.5 virtual CSTAResponseInfo ClearMessageWaiting (string *targetDevice*) [virtual]

Turns off the message waiting indicator on a device's display.

Parameters

targetDevice The device for which the indicator is to be turned off.

Returns

A CSTAResponseInfo object.

3.9.3.6 virtual CSTAResponseInfo ConsultationCall (ConnectionID *existingCall*, string *targetDevice*) [virtual]

Instruct the PBX to do a consultation call.

Parameters

existingCall The connection id of the call for which the consultation call will be made.

targetDevice Identifier (e.g., phone number) of the device that is the target of the consultation call.

Returns

A CSTAResponseInfo object.

3.9.3.7 virtual string EncodeAnswerCall (CSTAResponseInfo *response*, ConnectionID *callToAnswer*, string *deviceToLift*) [protected, virtual]

Encodes an AnswerCall message.

Parameters

response A CSTAResponseInfo object.

callToAnswer The ConnectionID of the call to answer.

deviceToLift The device (e.g., "800") that is to answer the call.

Returns

The encoded message.

3.9.3.8 virtual string EncodeAnswerCall (CSTARResponseInfo response, ConnectionID callToAnswer) [protected, virtual]

Encodes an AnswerCall message.

Parameters

response A CSTARResponseInfo object.

callToAnswer The ConnectionID of the call to answer.

Returns

The encoded message.

3.9.3.9 virtual string EncodeClearConnection (CSTARResponseInfo response, ConnectionID connectionToClear) [protected, virtual]

Encodes a ClearConnection message.

Parameters

response A CSTARResponseInfo object.

connectionToClear The ConnectionID of the connection to clear.

Returns

The encoded message.

3.9.3.10 virtual string EncodeConsultationCall (CSTARResponseInfo response, ConnectionID existingCall, string targetDevice) [protected, virtual]

Encodes a ConsultationCall message.

Parameters

response A CSTARResponseInfo object.

existingCall The connection id of the call for which the consultation call will be made.

targetDevice Identifier (e.g., phone number) of the device that is the target of the consultation call.

Returns

The encoded message.

**3.9.3.11 virtual string EncodeGetAgentState (CSTARResponseInfo *response*, string *targetDevice*)
[protected, virtual]**

Encodes a GetAgentState message.

Parameters

response A CSTARResponseInfo object.

targetDevice The device whose agent state is desired.

Returns

The encoded message.

3.9.3.12 virtual string EncodeGetDND (CSTARResponseInfo *response*, string *targetDevice*) [protected, virtual]

Encodes a GetDoNotDisturb message.

Parameters

response A CSTARResponseInfo object.

targetDevice The phone for which the Do Not Disturb setting is desired.

Returns

The encoded message.

**3.9.3.13 virtual string EncodeGetSFDevices (CSTARResponseInfo *response*, ReqDeviceCategory *category*)
[protected, virtual]**

Encodes a GetSwitchingFunctionDevices message.

Parameters

response A CSTARResponseInfo object.

category The category of device for which the list is desired.

Returns

The encoded message.

**3.9.3.14 virtual string EncodeHoldCall (CSTARResponseInfo *response*, ConnectionID *callToHold*)
[protected, virtual]**

Encodes a HoldCall message.

Parameters

response A CSTARResponseInfo object.

callToHold The ConnectionID object for the call to put on hold.

Returns

The encoded message.

3.9.3.15 virtual string EncodeInvokeID (string *initialMessage*) [virtual]

This method prepends an invoke ID to an already encoded XML CSTA message.

Parameters

initialMessage The XML CSTA message without the invoke ID.

Returns

The message with the invoke ID prepended.

3.9.3.16 virtual string EncodeMakeCall (CSTARResponseInfo *response*, string *callingDevice*, string *calledDevice*) [protected, virtual]

Encodes a MakeCall message.

Parameters

response A CSTARResponseInfo object.

callingDevice The device that is making the call.

calledDevice The device that is being called.

Returns

The encoded message.

3.9.3.17 virtual string EncodeMonitorStart (CSTARResponseInfo *response*, ConnectionID *targetCall*) [protected, virtual]

Encodes a MonitorStart message to monitor a call.

Parameters

response A CSTARResponseInfo object.

targetCall The call to monitor.

Returns

The encoded message.

3.9.3.18 virtual string EncodeMonitorStart (CSTARResponseInfo *response*, string *targetDevice*) [protected, virtual]

Encodes a MonitorStart message to monitor a device.

Parameters

response A CSTARResponseInfo object.

targetDevice Identifier (e.g., telephone number) of the device to monitor.

Returns

The encoded message.

3.9.3.19 virtual string EncodeMonitorStop (CSTARResponseInfo *response*, string *xref*) [protected, virtual]

Encodes a MonitorStop message.

Parameters

response A CSTARResponseInfo object.

xref The cross reference id of the monitor request as a MonitorCrossRefID object.

Returns

The encoded message.

3.9.3.20 virtual string EncodeRetrieveCall (CSTARResponseInfo *response*, ConnectionID *callToRetrieve*) [protected, virtual]

Encodes a RetrieveCall message.

Parameters

response A CSTARResponseInfo object.

callToRetrieve The ConnectionID of the call to retrieve.

Returns

The encoded message.

3.9.3.21 virtual string EncodeSendData (CSTARResponseInfo *response*, IOCrossRefID *xref*, string *strText*) [protected, virtual]

Encodes a SendData message.

Parameters

response A CSTARResponseInfo object.

xref An IOCrossRefID object, most likely obtained by a previous call to StartDataPath.

strText The text to send to the telephony device.

Returns

The encoded message.

3.9.3.22 virtual string EncodeSetAgentState (CSTARResponseInfo *response*, string *agentDevice*, ReqAgentState *agentState*) [protected, virtual]

Encodes a SetAgentState message.

Parameters

response A CSTARResponseInfo object.

agentDevice The device associated with the agent.

agentState An ReqAgentState object indicating the desired state of the agent.

Returns

The encoded message.

3.9.3.23 virtual string EncodeSetAgentState (CSTARResponseInfo response, string agentDevice, ReqAgentState agentState, string agentID) [protected, virtual]

Encodes a SetAgentState message.

Parameters

response A CSTARResponseInfo object.

agentDevice The device associated with the agent.

agentState An ReqAgentState object indicating the desired state of the agent.

agentID The agent id.

Returns

The encoded message.

3.9.3.24 virtual string EncodeSetDisplay (CSTARResponseInfo response, string targetDevice, string text) [protected, virtual]

Encodes a Set Display message.

Parameters

response A CSTARResponseInfo object.

targetDevice The device to which the text is to be sent.

text The text to be sent.

Returns

The encoded message.

3.9.3.25 virtual string EncodeSetDND (CSTARResponseInfo response, string targetDevice, bool dndOn) [protected, virtual]

Encodes a SetDoNotDisturb message.

Parameters

response A CSTARResponseInfo object.

targetDevice The device for which Do Not Disturb is to be set or cleared.

dndOn If true, indicates that Do Not Disturb is to be turned on. If false, indicates that Do Not Disturb is to be turned off.

Returns

The encoded message.

3.9.3.26 virtual string EncodeSetMsgWaiting (CSTARResponseInfo *response*, string *targetDevice*, bool *indicatorOn*) [protected, virtual]

Encodes a SetMessageWaiting message.

Parameters

response A CSTARResponseInfo object.

targetDevice The device for which the message waiting indicator is to be turned on or off.

indicatorOn If true, indicates that the message waiting indicator is to be turned on. If false, indicates that the message waiting indicator is to be turned off.

Returns

The encoded message.

3.9.3.27 virtual string EncodeSetRingerStatus (CSTARResponseInfo *response*, string *targetDevice*, string *targetRinger*, RingMode *rm*, long *ringPattern*) [protected, virtual]

Encodes a SetRingerStatus message.

Parameters

response A CSTARResponseInfo object.

targetDevice The device to ring.

targetRinger The id of the ringer to use for the ring. This argument can be specified as a character string (e.g. "abc"), a hex string (e.g. "010A05'H"), or a binary string (e.g. "000000010000101000000101'B").

rm A RingMode instance that indicates either ringing or notRinging.

ringPattern The indicator of the ring pattern to use.

Returns

The encoded message.

3.9.3.28 virtual string EncodeSingleStepTransfer (CSTARResponseInfo *response*, ConnectionID *callToTransfer*, string *transferToDevice*) [protected, virtual]

Encodes a SingleStepTransfer message.

Parameters

response A CSTARResponseInfo object.

callToTransfer A ConnectionID object that identifies the call to be transferred.

transferToDevice The device (e.g., "101") to which the call is to be transferred.

Returns

The encoded message.

**3.9.3.29 virtual string EncodeSnapshotCall (CSTARResponseInfo *response*, ConnectionID *callToSnapshot*)
[protected, virtual]**

Encodes a SnapshotCall message.

Parameters

response A CSTARResponseInfo object.

callToSnapshot The ConnectionID of the call for which the snapshot is desired.

Returns

The encoded message.

**3.9.3.30 virtual string EncodeSnapshotDevice (CSTARResponseInfo *response*, string *targetDevice*)
[protected, virtual]**

Encodes a SnapshotDevice message.

Parameters

response A CSTARResponseInfo object.

targetDevice Identifier (e.g., phone number) of the device for which the snapshot is desired.

Returns

The encoded message.

**3.9.3.31 virtual string EncodeStartDataPath (CSTARResponseInfo *response*, string *targetDevice*)
[protected, virtual]**

Encodes a StartDataPath message.

Parameters

response A CSTARResponseInfo object.

targetDevice The device against which a data path is to be started.

Returns

The encoded message.

**3.9.3.32 virtual string EncodeStartSession (CSTARResponseInfo *response*, string *applicationID*)
[protected, virtual]**

Encodes a StartApplicationSession message.

Parameters

response A CSTARResponseInfo object.

applicationID The application id to be encoded into the message.

Returns

The encoded message.

3.9.3.33 virtual string EncodeStopDataPath (CSTARResponseInfo *response*, IOCrossRefID *xref*) [protected, virtual]

Encodes a StopDataPath message.

Parameters

response A CSTARResponseInfo object.

xref An IOCrossRefID object, most likely obtained from a previous call to StartDataPath.

Returns

The encoded message.

3.9.3.34 virtual string EncodeStopSession (CSTARResponseInfo *response*) [protected, virtual]

Encodes a StopApplicationSession message.

Parameters

response A CSTARResponseInfo object.

Returns

The encoded message.

3.9.3.35 virtual string EncodeTransferCall (CSTARResponseInfo *response*, ConnectionID *initiatedCall*, ConnectionID *originalCall*) [protected, virtual]

Encodes a TransferCall message.

Parameters

response A CSTARResponseInfo object.

initiatedCall ConnectionID of the new call initiated by the consultation call. The initiatedCall member of the ConsultationCallResponse_ELEM class, for example, contains this ConnectionID.

originalCall ConnectionID of the original call. The somewhat confusingly named callingDevice member of the MakeCallResponse_ELEM class contains this ConnectionID, as does the establishedConnection member of the EstablishedEvent_ELEM class.

Returns

The encoded message.

3.9.3.36 virtual CSTARResponseInfo GetAgentState (string *agentDevice*) [virtual]

Gets the state of the agent associated with a device.

Parameters

agentDevice The device associated with the agent.

Returns

A CSTARResponseInfo object.

3.9.3.37 virtual CSTAResponseInfo GetDoNotDisturb (string *targetDevice*) [virtual]

Gets the Do Not Disturb setting for a phone.

Parameters

targetDevice The phone for which the Do Not Disturb setting is desired.

Returns

A CSTAResponseInfo object.

3.9.3.38 virtual CSTAResponseInfo GetSFDevices () [virtual]

Sends a Get Switching Function [Devices](#) request to the PBX.

Returns

A CSTAResponseInfo object.

3.9.3.39 virtual CSTAResponseInfo GetSFDevices (ReqDeviceCategory *deviceCategory*) [virtual]

Sends a Get Switching Function [Devices](#) request to the PBX.

Parameters

deviceCategory The category of device for which the list is desired.

Returns

A CSTAResponseInfo object.

3.9.3.40 virtual CSTAResponseInfo HoldCall (ConnectionID *callToHold*) [virtual]

Instruct the PBX to hold a call.

Parameters

callToHold The ConnectionID of the call to be held.

Returns

A CSTAResponseInfo object.

3.9.3.41 virtual CSTAResponseInfo MakeCall (string *callingDevice*, string *calledDevice*) [virtual]

Instruct the PBX to place a call.

Parameters

callingDevice Identifier (e.g., phone number) of the device making the call.

calledDevice Identifier (e.g., phone number) of the device being called.

Returns

A CSTAResponseInfo object.

3.9.3.42 virtual CSTAResponseInfo MonitorStart (ConnectionID *callToMonitor*) [virtual]

Issues a MonitorStart request to the PBX to monitor a call.

Parameters

callToMonitor The call to monitor.

Returns

A CSTAResponseInfo object.

3.9.3.43 virtual CSTAResponseInfo MonitorStart (string *deviceToMonitor*) [virtual]

Issues a MonitorStart request to the PBX to monitor a device.

Parameters

deviceToMonitor Identifier (e.g., telephone number) of the device to monitor.

Returns

A CSTAResponseInfo object.

Reimplemented in [VodiaSNOMOne](#).

3.9.3.44 virtual CSTAResponseInfo MonitorStop (string *crossRefID*) [virtual]

Stop a previously started PBX monitor request.

Parameters

crossRefID The cross reference id of the monitor request as a MonitorCrossRefID object.

Returns

A CSTAResponseInfo object.

3.9.3.45 virtual CSTAResponseInfo MonitorStopAtDevice (string *monitoredDevice*) [virtual]

This method stops all monitors active against the indicated device, regardless of what thread started the monitor. The method will only stop monitors started through the [MonitorStart\(\)](#) method.

Parameters

monitoredDevice The monitored device (e.g., extension).

Returns

If no problems are encountered, the method returns a CSTAResponseInfo object containing the response from the PBX for the LAST MonitorStop message.

If any problems are encountered, the method returns a CSTAResponseInfo object containing information about the error, including any response from the PBX for the problematic MonitorStop message.

3.9.3.46 virtual CSTAResponseInfo RetrieveCall (ConnectionID *callToRetrieve*) [virtual]

Retrieves a held call.

Parameters

callToRetrieve The ConnectionID of the call to retrieve.

Returns

A CSTAResponseInfo object.

3.9.3.47 virtual CSTAResponseInfo RingDevice (string *targetDevice*, string *targetRinger*, long *ringPattern*) [virtual]

Causes a telephony device to ring.

Parameters

targetDevice The device to ring.

targetRinger The id of the ringer to use for the ring. This argument can be specified as a character string (e.g, "abc"), a hex string (e.g, "010A05'H"), or a binary string (e.g, "000000010000101000000101'B").

ringPattern The indicator of the ring pattern to use.

Returns

A CSTAResponseInfo object.

3.9.3.48 virtual CSTAResponseInfo SendData (IOCrossRefID *xref*, string *text*) [virtual]

Sends a text message to a telephony device.

Parameters

xref An IOCrossRefID object, most likely obtained by a previous call to StartDataPath.

text The text to send to the telephony device.

Returns

A CSTAResponseInfo object.

3.9.3.49 virtual CSTAResponseInfo SetAgentState (string *agentDevice*, ReqAgentState *agentState*) [virtual]

Sets the state of an agent associated with a device.

Parameters

agentDevice The device associated with the agent.

agentState The desired state for the agent.

Returns

A CSTAResponseInfo object.

3.9.3.50 virtual CSTAResponseInfo SetAgentState (string *agentDevice*, ReqAgentState *agentState*, string *agentID*) [virtual]

Sets the state of an agent associated with a device.

Parameters

agentDevice The device associated with the agent.

agentState The desired state for the agent.

agentID The agent id.

Returns

A CSTAResponseInfo object.

3.9.3.51 virtual CSTAResponseInfo SetDisplay (string *targetDevice*, string *text*) [virtual]

Sends text to a telephony device's display

Parameters

targetDevice The device to which the text is to be sent.

text The text to be sent.

Returns

A CSTAResponseInfo object.

3.9.3.52 virtual CSTAResponseInfo SetDoNotDisturb (string *targetDevice*) [virtual]

Sets the Do Not Disturb feature for a phone.

Parameters

targetDevice The device for which Do Not Disturb is to be set.

Returns

A CSTAResponseInfo object.

3.9.3.53 virtual CSTAResponseInfo SetMessageWaiting (string *targetDevice*) [virtual]

Turns on the message waiting indicator on a device's display.

Parameters

targetDevice The device for which the indicator is to be turned on.

Returns

A CSTAResponseInfo object.

3.9.3.54 virtual CSTAResponseInfo SingleStepTransfer (ConnectionID *callToTransfer*, string *transferToDevice*) [virtual]

Perform a single step transfer.

Parameters

callToTransfer A ConnectionID object that identifies the call to be transferred.

transferToDevice The device (e.g., "101") to which the call is to be transferred.

Returns

A CSTAResponseInfo object.

3.9.3.55 virtual CSTAResponseInfo SnapshotCall (ConnectionID *callToSnapshot*) [virtual]

Instruct the PBX to take a snapshot of a call.

Parameters

callToSnapshot The ConnectionID of the call for which the snapshot is desired.

Returns

A CSTAResponseInfo object.

3.9.3.56 virtual CSTAResponseInfo SnapshotDevice (string *deviceToSnapshot*) [virtual]

Instruct the PBX to take a snapshot of calls active at a device.

Parameters

deviceToSnapshot Identifier (e.g., phone number) of the device for which the snapshot is desired.

Returns

A CSTAResponseInfo object.

3.9.3.57 virtual CSTAResponseInfo StartDataPath (string *targetDevice*) [virtual]

Opens up a data path to a specified device.

Parameters

targetDevice Specifies the device to which a data path is to be opened.

Returns

A CSTAResponseInfo object.

3.9.3.58 virtual CSTAResponseInfo StartSession (string *applicationID*) [virtual]

Establish a session with the PBX.

Parameters

applicationID A free text string to identify the application.

Returns

A CSTAResponseInfo object.

3.9.3.59 virtual CSTAResponseInfo StartSession () [virtual]

Establish a session with the PBX, using "CSTADLL" as the application identifier.

Returns

A CSTAResponseInfo object.

3.9.3.60 virtual CSTAResponseInfo StopDataPath (IOCrossRefID *xref*) [virtual]

Stops a previously established data path

Parameters

xref An IOCrossRefID object, most likely obtained from a previous call to StartDataPath.

Returns

A CSTAResponseInfo object.

3.9.3.61 virtual CSTAResponseInfo StopRing (string *targetDevice*, string *targetRinger*, long *ringPattern*) [virtual]

Stops a ringer on a telephony device.

Parameters

targetDevice The device for which the ringer is to stop.

targetRinger The id of the ringer to stop. This argument can be specified as a character string (e.g, "abc"), a hex string (e.g, "'010A05'H"), or a binary string (e.g, "'000000010000101000000101'B").

ringPattern The indicator of the ring pattern to stop.

Returns

A CSTAResponseInfo object.

3.9.3.62 virtual CSTAResponseInfo StopSession () [virtual]

Stops a session with a PBX. The TCP/IP connection to the PBX will be terminated.

Returns

A CSTAResponseInfo object.

3.9.3.63 virtual CSTAResponseInfo TransferCall (ConnectionID *initiatedCall*, ConnectionID *originalCall*) [virtual]

Transfers a call. A consultation call must be done before calling this method.

Parameters

initiatedCall ConnectionID of the new call initiated by the consultation call. The *initiatedCall* member of the *ConsultationCallResponse_ELEM* class, for example, contains this ConnectionID.

originalCall ConnectionID of the original call. The somewhat confusingly named *callingDevice* member of the *MakeCallResponse_ELEM* class contains this ConnectionID, as does the *establishedConnection* member of the *EstablishedEvent_ELEM* class.

Returns

A *CSTAResponseInfo* object.

3.9.4 Property Documentation

3.9.4.1 PBXSession SessionObject [get]

The *PBXSession* object associated with this instance.

3.9.4.2 CSTAContext ThreadContext [get]

The *CSTAContext* structure for this thread.

3.10 IETF_CSTAp1 Class Reference

Inherits [Com::Objsys::Csta::Phase1::GenericCSTAp1](#).

Inherited by [AlcatelOXO](#), [PhilipsSopho](#), [SiemensRealitis](#), and [TadiranCoral](#).

Public Member Functions

- [IETF_CSTAp1](#) ([PBXSession](#) sessionObject)
- [IETF_CSTAp1](#) (string pbxSystem, int port)

3.10.1 Detailed Description

Implements CSTA phase 1 operations using IETF encoding, which puts a two-byte length in front of the BER message.

3.10.2 Constructor & Destructor Documentation

3.10.2.1 IETF_CSTAp1 (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

pbxSystem Well-known name or IP address of the PBX.

port Port on which the PBX listens for CSTA messages.

3.10.2.2 IETF_CSTAp1 ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

sessionObject A [PBXSession](#) object.

3.11 IETF_CSTAp2 Class Reference

Inherits [Com::Objsys::Csta::Phase2::GenericCSTAp2](#).

Inherited by [Alcatel4400](#).

Public Member Functions

- [IETF_CSTAp2](#) ([PBXSession](#) sessionObject)
- [IETF_CSTAp2](#) (string pbxSystem, int port)

3.11.1 Detailed Description

Implements CSTA phase 2 operations using IETF encoding, which puts a two-byte length in front of the BER message.

3.11.2 Constructor & Destructor Documentation

3.11.2.1 IETF_CSTAp2 (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

pbxSystem Well-known name or IP address of the PBX.

port Port on which the PBX listens for CSTA messages.

3.11.2.2 IETF_CSTAp2 ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

sessionObject A [PBXSession](#) object.

3.12 IETF_CSTAp3 Class Reference

Inherits [Com::Objsys::Csta::Phase3::GenericCSTAp3](#).

Inherited by [PanasonicKXTDE](#), and [SiemensCap](#).

Public Member Functions

- [IETF_CSTAp3](#) ([PBXSession](#) sessionObject)
- [IETF_CSTAp3](#) (string pbxSystem, int port)

3.12.1 Detailed Description

Implements CSTA phase 3 operations using IETF encoding, which puts a two-byte length in front of the BER message.

3.12.2 Constructor & Destructor Documentation

3.12.2.1 IETF_CSTAp3 (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

pbxSystem Well-known name or IP address of the PBX.

port Port on which the PBX listens for CSTA messages.

3.12.2.2 IETF_CSTAp3 ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

sessionObject A [PBXSession](#) object.

3.13 LicenseException Class Reference

3.13.1 Detailed Description

Defines an exception that occurs while trying to find license information.

3.14 PanasonicKXTDA Class Reference

Inherits [Com::Objsys::Csta::Devices::PanasonicKXTDE](#).

Public Member Functions

- [PanasonicKXTDA](#) ([PBXSession](#) sessionObject)
- [PanasonicKXTDA](#) (string pbxSystem, int port)

3.14.1 Detailed Description

Implements CSTA phase 3 operations for the Panasonic KX-TDA PBX device.

3.14.2 Constructor & Destructor Documentation

3.14.2.1 [PanasonicKXTDA](#) (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
port Port on which the PBX listens for CSTA messages.

3.14.2.2 [PanasonicKXTDA](#) ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

- sessionObject* A [PBXSession](#) object.

3.15 PanasonicKXTDE Class Reference

Inherits [Com::Objsys::Csta::Phase3::IETF_CSTAp3](#).

Inherited by [PanasonicKXTDA](#), and [PanasonicNCP](#).

Public Types

- enum [DeviceDataTypes](#)

Public Member Functions

- [CSTAResponseInfo AcquireControlRight](#) (string targetDevice)
- [CSTAResponseInfo ClearMessageWaiting](#) (string originatingDevice, string targetDevice)
- [CSTAResponseInfo GetDeviceData](#) (string device, [DeviceDataTypes](#) eDataType)
- [CSTAResponseInfo GetGroupMembers](#) (string groupDevice)
- override [CSTAResponseInfo GetSFDevices](#) ()
- [PanasonicKXTDE](#) ([PBXSession](#) sessionObject)
- [PanasonicKXTDE](#) (string pbxSystem, int port)
- [CSTAResponseInfo PDFStart](#) (string targetDevice)
- [CSTAResponseInfo PDFStop](#) (string targetDevice)
- [CSTAResponseInfo ReleaseControlRight](#) (string targetDevice)
- [CSTAResponseInfo ResetDisplay](#) (string targetDevice)
- [CSTAResponseInfo SendKmeMessage](#) ([Asn1BerEncodeBuffer](#) encodeBuffer)
- [CSTAResponseInfo SetMessageWaiting](#) (string originatingDevice, string targetDevice)

Protected Member Functions

- override int [EncodeACSEConnectionRequest](#) ([CSTAResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer)
- override int [EncodeMakeCallRequest](#) ([CSTAResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, string callingDevice, string calledDevice)
- override int [EncodeSetDisplayRequest](#) ([CSTAResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, string targetDevice, string text)

3.15.1 Detailed Description

Implements CSTA phase 3 operations for the Panasonic KX-TDE PBX device.

3.15.2 Member Enumeration Documentation

3.15.2.1 enum DeviceDataTypes

Provides symbolic names for the types of device information that can be obtained from a Panasonic PBX.

3.15.3 Constructor & Destructor Documentation

3.15.3.1 PanasonicKXTDE (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

pbxSystem Well-known name or IP address of the PBX.

port Port on which the PBX listens for CSTA messages.

3.15.3.2 PanasonicKXTDE (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.15.4 Member Function Documentation

3.15.4.1 CSTAResponseInfo AcquireControlRight (string *targetDevice*)

Acquires the right to control a telephony device. This method will cause a "PDF (Physical Device Feature) Start" Escape message to be sent to the PBX.

Parameters

targetDevice The target telephony device.

Returns

A CSTAResponseInfo object.

3.15.4.2 CSTAResponseInfo ClearMessageWaiting (string *originatingDevice*, string *targetDevice*)

Turns off a device's message waiting indicator.

Parameters

originatingDevice The device that originated the call back request.

targetDevice The device for which the message waiting indicator is to be turned off.

Returns

A CSTAResponseInfo object.

3.15.4.3 override int EncodeACSEConnectionRequest (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*) [protected, virtual]

Encodes an ACSE Association Request message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented from [GenericCSTAp3](#).

3.15.4.4 **override int EncodeMakeCallRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string callingDevice, string calledDevice) [protected, virtual]**

Encodes a MakeCall message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

callingDevice Identifier (e.g., phone number) of the device making the call.

calledDevice Identifier (e.g., phone number) of the device being called.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented from [GenericCSTAp3](#).

3.15.4.5 **override int EncodeSetDisplayRequest (CSTAResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, string targetDevice, string text) [protected, virtual]**

Encodes a SetDisplay message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

targetDevice The device to which the text is to be sent.

text The text to be sent.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented from [GenericCSTAp3](#).

3.15.4.6 **CSTAResponseInfo GetDeviceData (string device, DeviceDataTypes eDataType)**

Gets information about a device.

Parameters

device The device (e.g., "101") about which the information is desired.

eDataType The type of data requested (must be from the DeviceDataTypes enum in this class).

Returns

3.15.4.7 CSTAResponseInfo GetGroupMembers (string groupDevice)

Gets the members associated with a group device (for example, the extensions associated with an incoming call distribution group device).

Parameters

groupDevice The group device designation (e.g, "601").

Returns

A CSTAResponseInfo object.

3.15.4.8 override CSTAResponseInfo GetSFDevices () [virtual]

Returns a list of station (i.e., telephone) devices known to the PBX by sending a Get Switching Function [Devices](#) message that specifies just station devices.

Returns

A CSTAResponseInfo object.

Reimplemented from [GenericCSTAp3](#).

3.15.4.9 CSTAResponseInfo PDFStart (string targetDevice)

Sends a "PDF (Physical Device Feature) Start" Escape message to the PBX. This method is a convenience method that does the same thing as the [AcquireControlRight\(\)](#) method.

Parameters

targetDevice The target telephony device.

Returns

A CSTAResponseInfo object.

3.15.4.10 CSTAResponseInfo PDFStop (string targetDevice)

Sends a "PDF (Physical Device Feature) Stop" Escape message to the PBX. This method is a convenience method that does the same thing as the [ReleaseControlRight\(\)](#) method.

Parameters

targetDevice The target telephony device.

Returns

A CSTAResponseInfo object.

3.15.4.11 CSTAResponseInfo ReleaseControlRight (string *targetDevice*)

Releases the right to control a telephony device. This method will cause a "PDF (Physical Device Feature) Stop" Escape message to be sent to the PBX.

Parameters

targetDevice The target telephony device.

Returns

A CSTAResponseInfo object.

3.15.4.12 CSTAResponseInfo ResetDisplay (string *targetDevice*)

Resets a telephony device's display

Parameters

targetDevice The device which is to be reset.

Returns

A CSTAResponseInfo object.

3.15.4.13 CSTAResponseInfo SendKmeMessage (Asn1BerEncodeBuffer *encodeBuffer*)

Sends a Panasonic-specific KME message to the PBX.

Parameters

encodeBuffer A BER encode buffer instance containing a completely encoded KME message. The message must include the KMESpecificPrivateData segment, the EscapeArgument, and the ROSE header.

Returns

3.15.4.14 CSTAResponseInfo SetMessageWaiting (string *originatingDevice*, string *targetDevice*)

Turns on a device's message waiting indicator.

Parameters

originatingDevice The device that originated the call back request.

targetDevice The device for which the message waiting indicator is to be turned on.

Returns

A CSTAResponseInfo object.

3.16 PanasonicNCP Class Reference

Inherits [Com::Objsys::Csta::Devices::PanasonicKXTDE](#).

Public Member Functions

- [PanasonicNCP](#) ([PBXSession](#) sessionObject)
- [PanasonicNCP](#) (string pbxSystem, int port)

3.16.1 Detailed Description

Implements CSTA phase 3 operations for the Panasonic NCP PBX device.

3.16.2 Constructor & Destructor Documentation

3.16.2.1 [PanasonicNCP](#) (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
port Port on which the PBX listens for CSTA messages.

3.16.2.2 [PanasonicNCP](#) ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

- sessionObject* A [PBXSession](#) object.

3.17 PBXSession Class Reference

Public Member Functions

- delegate void [AsyncCallback](#) ([PBXSession](#) sessionObject, byte[] asyncData)
- delegate void [AsyncExceptionCallback](#) ([PBXSession](#) sessionObject, [PBXSessionException](#) exception)
- void [Close](#) ([CSTAContext](#) threadContext)
- delegate void [ConnectionCallback](#) ([PBXSession](#) sessionObject)
- void [Open](#) ([CSTAContext](#) threadContext)
- [PBXSession](#) (string pbxSystem, int port)
- [SocketState](#) [SendACSEMessage](#) (byte[] message, int messageLength, Constants.ACSEMessageTypes messageType, [CSTAContext](#) threadContext)
- void [SendMessage](#) (string messageType, byte[] message, int messageLength, [CSTAContext](#) threadContext)
- void [SendMessage](#) (byte[] message, int messageLength, [CSTAContext](#) threadContext)
- void [SendXMLMessage](#) (string messageType, string strMessage, [CSTAContext](#) threadContext)
- void [SendXMLMessage](#) (string strMessage, [CSTAContext](#) threadContext)
- [SocketState](#) [SendXMLSession](#) (string strMessage, Constants.XMLSessionMessageTypes enmMessageType, [CSTAContext](#) threadContext)
- void [WaitForROSEResponse](#) ([CSTAContext](#) threadContext)
- void [WaitForXMLResponse](#) ([CSTAContext](#) threadContext)
- delegate void [XMLAsyncCallback](#) ([PBXSession](#) sessionObject, string message)

Properties

- AsyncCallback [ClientCallback](#) [get, set]
- ConnectionCallback [ConnectionLostCallback](#) [get, set]
- bool [DebugClientCallback](#) [get, set]
- bool [DebugMode](#) [get, set]
- AsyncExceptionCallback [ExceptionCallback](#) [get, set]
- int [MaxReceiveTimeout](#) [get, set]
- Constants.Encoding [MessageEncoding](#) [get, set]
- string [PBXSystem](#) [get]
- int [Port](#) [get]
- AsyncCallback [SystemStatusCallback](#) [get, set]
- XMLAsyncCallback [XMLClientCallback](#) [get, set]
- XMLAsyncCallback [XMLSystemStatusCallback](#) [get, set]

3.17.1 Detailed Description

This class manages communication with a PBX. One instance of this class should be created for each PBX with which a CSTADLL client application needs to exchange CSTA messages.

The CSTA worker classes (e.g., Alcatel4400, PanasonicNCP) hold a reference to a [PBXSession](#) object. If the constructor for the worker class that takes a PBX identification and a PBX port is used, a [PBXSession](#) object is created. Alternatively, the client application can create a [PBXSession](#) instance and pass a reference to the instance to the other worker class constructor signature.

Only one [PBXSession](#) instance for a PBX/port combination should be created. The behavior is undefined if multiple [PBXSession](#) instances are created for the same PBX and port.

3.17.2 Constructor & Destructor Documentation

3.17.2.1 PBXSession (string *pbxSystem*, int *port*)

Constructs a [PBXSession](#) object.

Parameters

- pbxSystem* The name or IP address of the PBX system.
- port* The port on the PBX system to which the client is connecting.

3.17.3 Member Function Documentation

3.17.3.1 delegate void AsyncCallback (PBXSession *sessionObject*, byte[] *asyncData*)

Declaration of a callback function to be invoked when an asynchronous message is received, such as from a monitor session.

Parameters

- sessionObject* The session object for the PBX that generated the asynchronous message.
- asyncData* The data received asynchronously from the PBX.

3.17.3.2 delegate void AsyncExceptionCallback (PBXSession *sessionObject*, PBXSessionException *exception*)

Declaration of a callback function to be invoked if a condition is encountered in the asynchronous I/O handler that would otherwise result in an exception being thrown. Note that in a couple of cases the asynchronous code will still throw an exception, even if this callback is defined.

Parameters

- sessionObject* The session object for the PBX that sent a packet that triggered an exception condition.
- exception* The [PBXSessionException](#) object that would have been thrown in the asynchronous I/O handling code if this callback were not defined.

3.17.3.3 void Close (CSTAContext *threadContext*)

Terminates the session to the PBX. This method can be used to terminate sessions with PBX devices that don't accept ACSE release association requests.

Parameters

- threadContext* The context object for the calling thread.

3.17.3.4 delegate void ConnectionCallback (PBXSession *sessionObject*)

Declaration of a callback function to be invoked if the connection to the PBX is lost.

Parameters

- sessionObject* The session object for the PBX whose connection was lost.

3.17.3.5 void Open (CSTAContext threadContext)

This method can be used to establish communication with a PBX device before any messages are actually sent to the device.

Parameters

threadContext The thread context object.

3.17.3.6 SocketState SendACSEMessage (byte[] message, int messageLength, Constants.ACSEMessageTypes messageType, CSTAContext threadContext)

This method sends an ACSE message (either Make Association or Release Association) to the PBX and receives the response. This operation is done synchronously. If the Make Association needs to be done (usually it does), it must be done before any threads for sending and receiving CSTA messages are started.

This method is only intended to be used by client code that encodes its own ACSEMakeAssociation or ACSEReleaseAssociation message. Most clients can probably use the MakeACSEAssociation() and ReleaseACSEAssociation() methods that are in each phase's helper classes.

Parameters

message An encoded ACSE Make Association or Release Association message.

messageLength The length of the encoded message.

messageType A constant telling whether the message is an ACSE Make Association or an ACSE Release Association.

threadContext The thread context object.

Returns

A populated [SocketState](#) instance.

3.17.3.7 void SendMessage (string messageType, byte[] message, int messageLength, CSTAContext threadContext)

This method sends a message to the PBX using TCP/IP.

Parameters

messageType A string token to help identify the message in the CSTADLL log file.

message Byte array containing the encoded message to send.

messageLength The length of the encoded message.

threadContext The thread context object.

3.17.3.8 void SendMessage (byte[] message, int messageLength, CSTAContext threadContext)

This method sends a message to the PBX using TCP/IP.

Parameters

message Byte array containing the encoded message to send.

messageLength The length of the encoded message.

threadContext The thread context object.

3.17.3.9 void SendXMLMessage (string *messageType*, string *strMessage*, CSTAContext *threadContext*)

This method sends an XML message to the PBX using TCP/IP.

Parameters

messageType A string token to help identify the message in the CSTADLL log file.

strMessage The XML message to send.

threadContext The thread context object.

3.17.3.10 void SendXMLMessage (string *strMessage*, CSTAContext *threadContext*)

This method sends an XML message to the PBX using TCP/IP.

Parameters

strMessage The XML message to send.

threadContext The thread context object.

3.17.3.11 SocketState SendXMLSession (string *strMessage*, Constants.XMLSessionMessageTypes *enmMessageType*, CSTAContext *threadContext*)

This method sends an XML session management (ECMA-354) message to the PBX.

Parameters

strMessage The text of the XML message to send.

enmMessageType A constant indicating what kind of session management message is being sent.

threadContext The thread context object.

Returns

A populated [SocketState](#) instance if the message is a StartSession message. Null if the message is StopSession or ResetSession.

3.17.3.12 void WaitForROSEResponse (CSTAContext *threadContext*)

This method waits for a response to a CSTA message sent with a ROSE header.

Parameters

threadContext The [CSTAContext](#) object associated with the calling thread.

3.17.3.13 void WaitForXMLResponse (CSTAContext *threadContext*)

This method waits for a response to an XML CSTA message.

Parameters

threadContext The [CSTAContext](#) object associated with the calling thread.

3.17.3.14 **delegate void XMLAsyncCallback (PBXSession *sessionObject*, string *message*)**

Declaration of a callback function to be invoked when an asynchronous XML message is received, such as from a monitor session.

Parameters

sessionObject The session object for the PBX that generated the asynchronous message.

message The text of the message received asynchronously from the PBX.

3.17.4 **Property Documentation**

3.17.4.1 **AsyncCallback ClientCallback [get, set]**

Holds a reference to an asynchronous callback function. This function will be invoked if data is received asynchronously from the PBX, such as from a monitor operation.

3.17.4.2 **ConnectionCallback ConnectionLostCallback [get, set]**

Holds a reference to an asynchronous callback function. This function will be invoked if the connection to the PBX is lost.

3.17.4.3 **bool DebugClientCallback [get, set]**

Enables easier debugging of asynchronous callback methods. If set to true, the read for the next message from the PBX won't occur until after the client callback method returns. Normally the read for the next message occurs before the client callback method is invoked. The default value for this property is false. This property must be set to false in order to send CSTA messages from a callback method.

3.17.4.4 **bool DebugMode [get, set]**

Enables behavior that facilitates debugging of the CSTADLL software. This property is most likely useful only to Objective Systems staff.

3.17.4.5 **AsyncExceptionCallback ExceptionCallback [get, set]**

Holds a reference to an asynchronous callback function. This function will be invoked if a condition occurs in the asynchronous I/O handler that otherwise would have resulted in an exception being thrown if this callback were not defined. Note that in a couple of cases the asynchronous code will still throw an exception, even if this callback is defined.

3.17.4.6 **int MaxReceiveTimeout [get, set]**

Specifies the amount of time, in milliseconds, to wait for a response to arrive from the PBX. The default value is 5,000 milliseconds (5 seconds).

3.17.4.7 **Constants.Encoding MessageEncoding [get, set]**

Indicates how messages exchanges with this PBX are encoded.

3.17.4.8 string PBXSystem [get]

The TCIP/IP address or well-known name of the PBX.

3.17.4.9 int Port [get]

The port where the PBX listens for CSTA messages.

3.17.4.10 AsyncCallback SystemStatusCallback [get, set]

Holds a reference to an asynchronous callback function. This function will be invoked if a BER System Status message is received asynchronously from the PBX. The client application does NOT need to send a System Status Response, since that is taken care of by CSTADLL.

3.17.4.11 XMLAsyncCallback XMLClientCallback [get, set]

Holds a reference to an asynchronous XML callback function. This function will be invoked if an XML message is received asynchronously from the PBX, such as from a monitor operation.

3.17.4.12 XMLAsyncCallback XMLSystemStatusCallback [get, set]

Holds a reference to an asynchronous callback function. This function will be invoked if an XML System Status message is received asynchronously from the PBX. The client application does NOT need to send a System Status Response, since that is taken care of by CSTADLL.

3.18 PBXSessionException Class Reference

3.18.1 Detailed Description

Defines an exception that occurs while communicating with a PBX.

3.19 PBXSessionHelper Class Reference

Properties

- static string [LoggingFolder](#) [get, set]

3.19.1 Detailed Description

This class holds static properties that affect all PBX sessions.

3.19.2 Property Documentation

3.19.2.1 string [LoggingFolder](#) [static, get, set]

Specifies a folder to receive the log file. If not specified, the log file will go into whatever folder the calling .exe resides in.

3.20 Phase1Opcodes Class Reference

Public Types

- enum [Opcodes](#)

3.20.1 Detailed Description

This class contains a public enum that contains symbolic names for the opcodes that define CSTA phase 1 operations.

3.20.2 Member Enumeration Documentation

3.20.2.1 enum Opcodes

Contains symbolic names for the opcodes that define CSTA phase 1 operations.

3.21 Phase2Opcodes Class Reference

Public Types

- enum [Opcodes](#)

3.21.1 Detailed Description

This class contains a public enum that contains symbolic names for the opcodes that define CSTA phase 2 operations.

3.21.2 Member Enumeration Documentation

3.21.2.1 enum Opcodes

Contains symbolic names for the opcodes that define CSTA phase 2 operations.

3.22 Phase3Opcodes Class Reference

Public Types

- enum [Opcodes](#)

3.22.1 Detailed Description

This class contains a public enum that contains symbolic names for the opcodes that define CSTA phase 3 operations.

3.22.2 Member Enumeration Documentation

3.22.2.1 enum Opcodes

Contains symbolic names for the opcodes that define CSTA phase 3 operations.

3.23 PhilipsSopho Class Reference

Inherits [Com::Objsys::Csta::Phase1::IETF_CSTAp1](#).

Public Member Functions

- [CSTARResponseInfo MakeACSEAssociation](#) (bool testMode)
- override [CSTARResponseInfo MakeACSEAssociation](#) ()
- [PhilipsSopho](#) ([PBXSession](#) sessionObject)
- [PhilipsSopho](#) (string pbxSystem, int port)

3.23.1 Detailed Description

Implements CSTA phase 1 operations for the Philips Sopho PBX device.

3.23.2 Constructor & Destructor Documentation

3.23.2.1 PhilipsSopho (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

pbxSystem Well-known name or IP address of the PBX.

port Port on which the PBX listens for CSTA messages.

3.23.2.2 PhilipsSopho ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

sessionObject A [PBXSession](#) object.

3.23.3 Member Function Documentation

3.23.3.1 [CSTARResponseInfo MakeACSEAssociation](#) (bool *testMode*)

Establish an ACSE association with the PBX.

Parameters

testMode True to send a test mode message, false to send a licensed message.

Returns

A [CSTARResponseInfo](#) object.

3.23.3.2 **override CSTAResponseInfo MakeACSEAssociation () [virtual]**

Establish a licensed (i.e., not test mode) ACSE association with the PBX.

Returns

A CSTAResponseInfo object.

Reimplemented from [GenericCSTAp1](#).

3.24 ResetSessionInfo Class Reference

3.24.1 Detailed Description

This class provides information that needs to be passed to the thread that periodically sends an XML ResetSession message to the PBX.

3.25 ROSEParseInfo Class Reference

3.25.1 Detailed Description

Contains information about a received CSTA message obtained by decoding the message's ROSE header.

3.26 SiemensCap Class Reference

Inherits [Com::Objsys::Csta::Phase3::IETF_CSTAp3](#).

Inherited by [SiemensHipath4000](#).

Public Member Functions

- [CSTAResponseInfo MakeACSEAssociation](#) (string appid, string login, string passwd, bool nativeMode, CSTAVersion cv)
- [CSTAResponseInfo MakeACSEAssociation](#) (string appid, string login, string passwd, bool nativeMode)
- [CSTAResponseInfo MakeACSEAssociation](#) (string login, string passwd)
- [SiemensCap](#) ([PBXSession](#) sessionObject)
- [SiemensCap](#) (string pbxSystem, int port)

Protected Member Functions

- override int [EncodeACSEConnectionRequest](#) ([CSTAResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer)

3.26.1 Detailed Description

Implements CSTA phase 3 operations for the Siemens CAP PBX device.

3.26.2 Constructor & Destructor Documentation

3.26.2.1 [SiemensCap](#) (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

pbxSystem Well-known name or IP address of the PBX.

port Port on which the PBX listens for CSTA messages.

3.26.2.2 [SiemensCap](#) ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

sessionObject A [PBXSession](#) object.

3.26.3 Member Function Documentation

3.26.3.1 override int [EncodeACSEConnectionRequest](#) ([CSTAResponseInfo](#) *response*, [Asn1BerEncodeBuffer](#) *encodeBuffer*) [**protected**, **virtual**]

Encodes an ACSE Association Request message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented from [GenericCSTAp3](#).

3.26.3.2 CSTAResponseInfo MakeACSEAssociation (string *appid*, string *login*, string *passwd*, bool *nativeMode*, CSTAVersion *cv*)

Establish an ACSE association with the PBX using the specified arguments.

Parameters

appid The application id to be encoded into the ACSE Association Request.

login The login to be encoded into the ACSE Association Request.

passwd The password to be encoded into the ACSE Association Request.

nativeMode If true, "native=true" will be included in the ACSE Association Request. If false, "native=true" will not be included.

cv A CSTAVersion object representing the CSTA version to encode into the ACSE Association Request.

Returns

A CSTAResponseInfo object.

3.26.3.3 CSTAResponseInfo MakeACSEAssociation (string *appid*, string *login*, string *passwd*, bool *nativeMode*)

Establish an ACSE association with the PBX using the specified arguments.

Parameters

appid The application id to be encoded into the ACSE Association Request.

login The login to be encoded into the ACSE Association Request.

passwd The password to be encoded into the ACSE Association Request.

nativeMode If true, "native=true" will be included in the ACSE Association Request. If false, "native=true" will not be included.

Returns

A CSTAResponseInfo object.

3.26.3.4 CSTAResponseInfo MakeACSEAssociation (string *login*, string *passwd*)

Establish an ACSE association with the PBX using a specified login and password.

Parameters

login The login to be encoded into the ACSE Association Request.

passwd The password to be encoded into the ACSE Association Request.

Returns

A CSTAResponseInfo object.

3.27 SiemensHicom300 Class Reference

Inherits [Com::Objsys::Csta::Phase1::GenericCSTAp1](#).

Public Member Functions

- [SiemensHicom300](#) ([PBXSession](#) sessionObject)
- [SiemensHicom300](#) (string pbxSystem, int port)

3.27.1 Detailed Description

Implements CSTA phase 1 operations for the Siemens Hicom 300 PBX device.

3.27.2 Constructor & Destructor Documentation

3.27.2.1 SiemensHicom300 (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
port Port on which the PBX listens for CSTA messages.

3.27.2.2 SiemensHicom300 ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

- sessionObject* A [PBXSession](#) object.

3.28 SiemensHipath3000p2 Class Reference

Inherits [Com::Objsys::Csta::Phase2::GenericCSTAp2](#).

Public Member Functions

- [SiemensHipath3000p2](#) ([PBXSession](#) sessionObject)
- [SiemensHipath3000p2](#) (string pbxSystem, int port)

3.28.1 Detailed Description

Implements CSTA phase 2 operations for the Siemens Hipath 3000 PBX device.

3.28.2 Constructor & Destructor Documentation

3.28.2.1 SiemensHipath3000p2 (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
port Port on which the PBX listens for CSTA messages.

3.28.2.2 SiemensHipath3000p2 ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

- sessionObject* A [PBXSession](#) object.

3.29 SiemensHipath3000p3 Class Reference

Inherits [Com::Objsys::Csta::Phase3::GenericCSTAp3](#).

Inherited by [UnifyOpenscapeX5](#).

Public Member Functions

- virtual [CSTAResponseInfo MakeACSEAssociation](#) (string userName, string password)
- [SiemensHipath3000p3 \(PBXSession sessionObject\)](#)
- [SiemensHipath3000p3](#) (string pbxSystem, int port)

Protected Member Functions

- override int [EncodeACSEConnectionRequest](#) ([CSTAResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer)
- override int [EncodeSingleStepTransferRequest](#) ([CSTAResponseInfo](#) response, [Asn1BerEncodeBuffer](#) encodeBuffer, [SingleStepTransferInfo](#) sstInfo)

3.29.1 Detailed Description

Implements CSTA phase 3 operations for the Siemens Hipath 3000 PBX device.

3.29.2 Constructor & Destructor Documentation

3.29.2.1 [SiemensHipath3000p3](#) (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
- port* Port on which the PBX listens for CSTA messages.

3.29.2.2 [SiemensHipath3000p3](#) ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

- sessionObject* A [PBXSession](#) object.

3.29.3 Member Function Documentation

3.29.3.1 `override int EncodeACSEConnectionRequest` ([CSTAResponseInfo](#) *response*, [Asn1BerEncodeBuffer](#) *encodeBuffer*) [`protected`, `virtual`]

Encodes an ACSE Association Request message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented from [GenericCSTAp3](#).

3.29.3.2 `override int EncodeSingleStepTransferRequest (CSTARResponseInfo response, Asn1BerEncodeBuffer encodeBuffer, SingleStepTransferInfo sstInfo) [protected, virtual]`

Encodes a SingleStepTransfer message.

Parameters

response A CSTA ResponseInfo object.

encodeBuffer An encode buffer object into which the message will be encoded.

sstInfo A SingleStepTransferInfo object.

Returns

The length of the encoded message, or -1 if an error occurred.

Reimplemented from [GenericCSTAp3](#).

3.29.3.3 `virtual CSTARResponseInfo MakeACSEAssociation (string userName, string password) [virtual]`

Establish an ACSE association with the PBX.

Parameters

userName The user name to send to the PBX.

password The password to send to the PBX.

Returns

A CSTARResponseInfo object.

3.30 SiemensHipath4000 Class Reference

Inherits [Com::Objsys::Csta::Devices::SiemensCap](#).

Public Member Functions

- [SiemensHipath4000](#) ([PBXSession](#) sessionObject)
- [SiemensHipath4000](#) (string pbxSystem, int port)

3.30.1 Detailed Description

Implements CSTA phase 3 operations for the Siemens Hipath 4000 PBX device.

3.30.2 Constructor & Destructor Documentation

3.30.2.1 [SiemensHipath4000](#) (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
port Port on which the PBX listens for CSTA messages.

3.30.2.2 [SiemensHipath4000](#) ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

- sessionObject* A [PBXSession](#) object.

3.31 SiemensRealitis Class Reference

Inherits [Com::Objsys::Csta::Phase1::IETF_CSTAp1](#).

Public Member Functions

- [SiemensRealitis](#) ([PBXSession](#) sessionObject)
- [SiemensRealitis](#) (string pbxSystem, int port)

3.31.1 Detailed Description

Implements CSTA phase 1 operations for the Siemens Realitis PBX device.

3.31.2 Constructor & Destructor Documentation

3.31.2.1 SiemensRealitis (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
port Port on which the PBX listens for CSTA messages.

3.31.2.2 SiemensRealitis ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

- sessionObject* A [PBXSession](#) object.

3.32 SingleStepTransferInfo Class Reference

Properties

- string [TransferFromDevice](#) [get, set]
- string [TransferringCallID](#) [get, set]
- string [TransferToDevice](#) [get, set]

3.32.1 Detailed Description

Contains information needed to complete a phase 3 single step transfer request.

3.32.2 Property Documentation

3.32.2.1 string TransferFromDevice [get, set]

Identification (e.g., phone number) of the device from which the call is being transferred.

3.32.2.2 string TransferringCallID [get, set]

The call id number associated with the device from which the call is being transferred.

3.32.2.3 string TransferToDevice [get, set]

Identification (e.g., phone number) of the device to which the call is being transferred.

3.33 SocketState Class Reference

Properties

- byte[] [AckBuffer](#) [get, set]
- byte[] [ReadBuffer](#) [get, set]
- List< byte[] > [ReadBuffers](#) [get, set]
- int [TotalLength](#) [get, set]

3.33.1 Detailed Description

This class contains the response received from the PBX and state information about the exchange with the PBX that is used internally by CSTADLL.

3.33.2 Property Documentation

3.33.2.1 byte [] AckBuffer [get, set]

Contains the first response from the PBX for situations where the PBX sends multiple response messages (e.g., Get Switching Function [Devices](#)). The data messages that are sent after this ack will be in ReadBuffers.

3.33.2.2 byte [] ReadBuffer [get, set]

Contains the bytes most recently read from the socket. This buffer will be filled in bit by bit as the message is read.

3.33.2.3 List<byte[]> ReadBuffers [get, set]

Contains multiple collections of bytes read from the socket. This array is used for situations where a response to a message comes in multiple segments (e.g., Get Switching Function [Devices](#)). For these situations the immediate response will be in AckBuffer.

3.33.2.4 int TotalLength [get, set]

The total length of a complete message received from the PBX. This is also used as an offset into the read buffer so we can build the message as it's received.

3.34 TadiranCoral Class Reference

Inherits [Com::Objsys::Csta::Phase1::IETF_CSTAp1](#).

Public Member Functions

- [TadiranCoral](#) ([PBXSession](#) sessionObject)
- [TadiranCoral](#) (string pbxSystem, int port)

3.34.1 Detailed Description

Implements CSTA phase 1 operations for the Tadiran Coral PBX device.

3.34.2 Constructor & Destructor Documentation

3.34.2.1 TadiranCoral (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
port Port on which the PBX listens for CSTA messages.

3.34.2.2 TadiranCoral ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

- sessionObject* A [PBXSession](#) object.

3.35 TransferCallInfo Class Reference

Properties

- string [ActiveDeviceCallID](#) [get, set]
- string [ActiveDeviceNumber](#) [get, set]
- string [ConnectedDevice](#) [get, set]
- string [HeldDevice](#) [get, set]
- string [HeldDeviceCallID](#) [get, set]
- string [HeldDeviceNumber](#) [get, set]

3.35.1 Detailed Description

Contains information needed to complete a phase 3 transfer call request.

3.35.2 Property Documentation

3.35.2.1 string ActiveDeviceCallID [get, set]

The call id associated with the device to which the call is being transferred.

3.35.2.2 string ActiveDeviceNumber [get, set]

The phone number to which the call is being transferred. This number is not necessarily the same as the value for ConnectedDevice.

3.35.2.3 string ConnectedDevice [get, set]

Identification (e.g., phone number) of the device to which the call is being transferred.

3.35.2.4 string HeldDevice [get, set]

Identification (e.g., phone number) of the device from which the call is being transferred.

3.35.2.5 string HeldDeviceCallID [get, set]

The call id associated with the device from which the call is being transferred.

3.35.2.6 string HeldDeviceNumber [get, set]

The phone number from which the call is being transferred. This number is not necessarily the same as the value for HeldDevice.

3.36 UnifyOpenscapeVoice Class Reference

Inherits [Com::Objsys::Csta::Xml::GenericXML](#).

Public Member Functions

- [UnifyOpenscapeVoice](#) ([PBXSession](#) sessionObject)
- [UnifyOpenscapeVoice](#) (string pbxSystem, int port)

3.36.1 Detailed Description

Implements CSTA XML operations for the Unify Openscape Voice PBX device.

3.36.2 Constructor & Destructor Documentation

3.36.2.1 UnifyOpenscapeVoice (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
port Port on which the PBX listens for CSTA messages.

3.36.2.2 UnifyOpenscapeVoice ([PBXSession](#) *sessionObject*)

Constructs an instance associated with the given [PBXSession](#) object.

Parameters

- sessionObject* A [PBXSession](#) object.

3.37 UnifyOpenscapeX5 Class Reference

Inherits [Com::Objsys::Csta::Devices::SiemensHipath3000p3](#).

Public Member Functions

- [UnifyOpenscapeX5 \(PBXSession sessionObject\)](#)
- [UnifyOpenscapeX5 \(string pbxSystem, int port\)](#)

3.37.1 Detailed Description

Implements CSTA phase 3 operations for the Unify Openscape Business X5 PBX device.

3.37.2 Constructor & Destructor Documentation

3.37.2.1 UnifyOpenscapeX5 (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

- pbxSystem* Well-known name or IP address of the PBX.
- port* Port on which the PBX listens for CSTA messages.

3.37.2.2 UnifyOpenscapeX5 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

- sessionObject* A PBXSession object.

3.38 VodiaSNOMOne Class Reference

Inherits [Com::Objsys::Csta::Xml::GenericXML](#).

Public Member Functions

- override [CSTAResponseInfo MonitorStart](#) (string deviceToMonitor)
- [CSTAResponseInfo StartSession](#) (string applicationID, string userName, string password)
- [CSTAResponseInfo StartSession](#) (string userName, string password)
- [VodiaSNOMOne \(PBXSession sessionObject\)](#)
- [VodiaSNOMOne](#) (string pbxSystem, int port)

3.38.1 Detailed Description

Implements CSTA XML operations for the Vodia SNOMOne PBX device.

3.38.2 Constructor & Destructor Documentation

3.38.2.1 VodiaSNOMOne (string *pbxSystem*, int *port*)

Constructs an instance associated with the given PBX identifier and port.

Parameters

pbxSystem Well-known name or IP address of the PBX.

port Port on which the PBX listens for CSTA messages.

3.38.2.2 VodiaSNOMOne (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.38.3 Member Function Documentation

3.38.3.1 override CSTAResponseInfo MonitorStart (string *deviceToMonitor*) [virtual]

Issues a MonitorStart request to the PBX.

Parameters

deviceToMonitor Identifier (e.g., telephone number) of the device to monitor.

Returns

A CSTAResponseInfo object.

Reimplemented from [GenericXML](#).

3.38.3.2 CSTAResponseInfo StartSession (string *applicationID*, string *userName*, string *password*)

Establish a session with the PBX.

Parameters

applicationID A free text string to identify the application.

userName The user name to send to the PBX.

password The password for the specified user name.

Returns

A CSTAResponseInfo object.

3.38.3.3 CSTAResponseInfo StartSession (string *userName*, string *password*)

Establish a session with the PBX, using "CSTADLL" as the application identifier.

Parameters

userName The user name to send to the PBX.

password The password for the specified user name.

Returns

A CSTAResponseInfo object.

Index

- AckBuffer
 - Com::Objsys::Csta::Common::SocketState, 127
- AcquireControlRight
 - Com::Objsys::Csta::Devices::PanasonicKXTDE, 97
- ActiveDeviceCallID
 - Com::Objsys::Csta::Phase3::TransferCallInfo, 129
- ActiveDeviceNumber
 - Com::Objsys::Csta::Phase3::TransferCallInfo, 129
- Alcatel4400
 - Com::Objsys::Csta::Devices::Alcatel4400, 9
- AlcatelOXE
 - Com::Objsys::Csta::Devices::AlcatelOXE, 11
- AlcatelOXO
 - Com::Objsys::Csta::Devices::AlcatelOXO, 12
- AlternateCall
 - Com::Objsys::Csta::Phase3::GenericCSTAp3, 50
- AnswerCall
 - Com::Objsys::Csta::Phase1::GenericCSTAp1, 19
 - Com::Objsys::Csta::Phase2::GenericCSTAp2, 34
 - Com::Objsys::Csta::Phase3::GenericCSTAp3, 50
 - Com::Objsys::Csta::Xml::GenericXML, 74
- AsyncCallback
 - Com::Objsys::Csta::Common::PBXSession, 103
- AsyncExceptionCallback
 - Com::Objsys::Csta::Common::PBXSession, 103
- ClearConnection
 - Com::Objsys::Csta::Phase1::GenericCSTAp1, 19
 - Com::Objsys::Csta::Phase2::GenericCSTAp2, 34
 - Com::Objsys::Csta::Phase3::GenericCSTAp3, 50
 - Com::Objsys::Csta::Xml::GenericXML, 74
- ClearDoNotDisturb
 - Com::Objsys::Csta::Phase1::GenericCSTAp1, 20
 - Com::Objsys::Csta::Phase2::GenericCSTAp2, 35
 - Com::Objsys::Csta::Phase3::GenericCSTAp3, 51
 - Com::Objsys::Csta::Xml::GenericXML, 75
- ClearMessageWaiting
 - Com::Objsys::Csta::Devices::PanasonicKXTDE, 97
 - Com::Objsys::Csta::Phase1::GenericCSTAp1, 20
 - Com::Objsys::Csta::Phase2::GenericCSTAp2, 35
 - Com::Objsys::Csta::Phase3::GenericCSTAp3, 51
 - Com::Objsys::Csta::Xml::GenericXML, 75
- ClientCallback
 - Com::Objsys::Csta::Common::PBXSession, 106
- Close
 - Com::Objsys::Csta::Common::PBXSession, 103
 - Com.Objsys.Csta.Common, 3
 - Com.Objsys.Csta.Devices, 4
 - Com.Objsys.Csta.Phase1, 5
 - Com.Objsys.Csta.Phase2, 6
 - Com.Objsys.Csta.Phase3, 7
 - Com.Objsys.Csta.Xml, 8
 - Com::Objsys::Csta::Common::CSTAContext, 14
 - ResponseFromPBX, 14
 - ResponsesFromPBX, 14
 - XMLResponseFromPBX, 14
 - XMLResponsesFromPBX, 14
 - Com::Objsys::Csta::Common::CSTARResponseInfo, 15
 - ResponseFromPBX, 15
 - ResponsesFromPBX, 15
 - StatusCode, 15
 - StatusMessage, 15
 - XMLResponseFromPBX, 15
 - XMLResponsesFromPBX, 16
 - Com::Objsys::Csta::Common::LicenseException, 94
 - Com::Objsys::Csta::Common::PBXSession, 102
 - AsyncCallback, 103
 - AsyncExceptionCallback, 103
 - ClientCallback, 106
 - Close, 103
 - ConnectionCallback, 103
 - ConnectionLostCallback, 106
 - DebugClientCallback, 106
 - DebugMode, 106
 - ExceptionCallback, 106
 - MaxReceiveTimeout, 106
 - MessageEncoding, 106
 - Open, 103
 - PBXSession, 103
 - PBXSystem, 106
 - Port, 107
 - SendACSEMessage, 104
 - SendMessage, 104
 - SendXMLMessage, 104, 105
 - SendXMLSession, 105
 - SystemStatusCallback, 107
 - WaitForROSEResponse, 105
 - WaitForXMLResponse, 105
 - XMLAsyncCallback, 105
 - XMLClientCallback, 107

XMLSystemStatusCallback, [107](#)
 Com::Objsys::Csta::Common::PBXSessionException, [108](#)
 Com::Objsys::Csta::Common::PBXSessionHelper, [109](#)
 LoggingFolder, [109](#)
 Com::Objsys::Csta::Common::ResetSessionInfo, [115](#)
 Com::Objsys::Csta::Common::ROSEParseInfo, [116](#)
 Com::Objsys::Csta::Common::SocketState, [127](#)
 AckBuffer, [127](#)
 ReadBuffer, [127](#)
 ReadBuffers, [127](#)
 TotalLength, [127](#)
 Com::Objsys::Csta::Devices::Alcatel4400, [9](#)
 Alcatel4400, [9](#)
 EncodeACSEConnectionRequest, [10](#)
 EncodeMakeCallRequest, [10](#)
 MakeACSEAssociation, [10](#)
 Com::Objsys::Csta::Devices::AlcatelOXE, [11](#)
 AlcatelOXE, [11](#)
 Com::Objsys::Csta::Devices::AlcatelOXO, [12](#)
 AlcatelOXO, [12](#)
 EncodeACSEConnectionRequest, [12](#)
 MakeACSEAssociation, [13](#)
 Com::Objsys::Csta::Devices::PanasonicKXTDA, [95](#)
 PanasonicKXTDA, [95](#)
 Com::Objsys::Csta::Devices::PanasonicKXTDE, [96](#)
 AcquireControlRight, [97](#)
 ClearMessageWaiting, [97](#)
 DeviceDataTypes, [96](#)
 EncodeACSEConnectionRequest, [97](#)
 EncodeMakeCallRequest, [98](#)
 EncodeSetDisplayRequest, [98](#)
 GetDeviceData, [98](#)
 GetGroupMembers, [99](#)
 GetSFDevices, [99](#)
 PanasonicKXTDE, [97](#)
 PDFStart, [99](#)
 PDFStop, [99](#)
 ReleaseControlRight, [99](#)
 ResetDisplay, [100](#)
 SendKmeMessage, [100](#)
 SetMessageWaiting, [100](#)
 Com::Objsys::Csta::Devices::PanasonicNCP, [101](#)
 PanasonicNCP, [101](#)
 Com::Objsys::Csta::Devices::PhilipsSopho, [113](#)
 MakeACSEAssociation, [113](#)
 PhilipsSopho, [113](#)
 Com::Objsys::Csta::Devices::SiemensCap, [117](#)
 EncodeACSEConnectionRequest, [117](#)
 MakeACSEAssociation, [118](#)
 SiemensCap, [117](#)
 Com::Objsys::Csta::Devices::SiemensHicom300, [120](#)
 SiemensHicom300, [120](#)
 Com::Objsys::Csta::Devices::SiemensHipath3000p2, [121](#)
 SiemensHipath3000p2, [121](#)
 Com::Objsys::Csta::Devices::SiemensHipath3000p3, [122](#)
 EncodeACSEConnectionRequest, [122](#)
 EncodeSingleStepTransferRequest, [123](#)
 MakeACSEAssociation, [123](#)
 SiemensHipath3000p3, [122](#)
 Com::Objsys::Csta::Devices::SiemensHipath4000, [124](#)
 SiemensHipath4000, [124](#)
 Com::Objsys::Csta::Devices::SiemensRealitis, [125](#)
 SiemensRealitis, [125](#)
 Com::Objsys::Csta::Devices::TadiranCoral, [128](#)
 TadiranCoral, [128](#)
 Com::Objsys::Csta::Devices::UnifyOpenscapeVoice, [130](#)
 UnifyOpenscapeVoice, [130](#)
 Com::Objsys::Csta::Devices::UnifyOpenscapeX5, [131](#)
 UnifyOpenscapeX5, [131](#)
 Com::Objsys::Csta::Devices::VodiaSNOMOne, [132](#)
 MonitorStart, [132](#)
 StartSession, [132](#), [133](#)
 VodiaSNOMOne, [132](#)
 Com::Objsys::Csta::Phase1::GenericCSTAp1, [17](#)
 AnswerCall, [19](#)
 ClearConnection, [19](#)
 ClearDoNotDisturb, [20](#)
 ClearMessageWaiting, [20](#)
 ConsultationCall, [20](#)
 DivertCall, [20](#)
 EncodeACSEConnectionRequest, [21](#)
 EncodeACSEReleaseRequest, [21](#)
 EncodeAnswerCallRequest, [21](#), [22](#)
 EncodeClearConnectionRequest, [22](#)
 EncodeConsultationCallRequest, [22](#)
 EncodeDivertCallRequest, [23](#)
 EncodeMakeCallRequest, [23](#)
 EncodeMonitorStartRequest, [23](#), [24](#)
 EncodeMonitorStopRequest, [24](#)
 EncodeQueryDeviceRequest, [24](#)
 EncodeRetrieveCallRequest, [25](#)
 EncodeROSERequestHeader, [25](#)
 EncodeSetOrClearDNDRequest, [25](#)
 EncodeSetOrClearMWRequest, [26](#)
 EncodeSnapshotDeviceRequest, [26](#)
 EncodeTransferCallRequest, [26](#), [27](#)
 GenericCSTAp1, [18](#)
 MakeACSEAssociation, [27](#)
 MakeCall, [27](#)
 MonitorStart, [27](#), [28](#)
 MonitorStop, [28](#)
 QueryDevice, [29](#)
 ReleaseACSEAssociation, [29](#)
 RetrieveCall, [29](#)
 SessionObject, [31](#)
 SetDoNotDisturb, [29](#)
 SetMessageWaiting, [29](#)

SnapshotDevice, 30
 ThreadContext, 31
 TransferCall, 30
 Com::Objsys::Csta::Phase1::IETF_CSTAp1, 91
 IETF_CSTAp1, 91
 Com::Objsys::Csta::Phase1::Phase1Opcodes, 110
 Opcodes, 110
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 32
 AnswerCall, 34
 ClearConnection, 34
 ClearDoNotDisturb, 35
 ClearMessageWaiting, 35
 ConsultationCall, 35
 DivertCall, 35
 EncodeACSEConnectionRequest, 36
 EncodeACSEReleaseRequest, 36
 EncodeAnswerCallRequest, 36, 37
 EncodeClearConnectionRequest, 37
 EncodeConsultationCallRequest, 37
 EncodeDivertCallRequest, 38
 EncodeMakeCallRequest, 38
 EncodeMonitorStartRequest, 38, 39
 EncodeMonitorStopRequest, 39
 EncodeQueryDeviceRequest, 40
 EncodeRetrieveCallRequest, 40
 EncodeROSERequestHeader, 40
 EncodeSetOrClearDNDRequest, 40
 EncodeSetOrClearMWRequest, 41
 EncodeSnapshotDeviceRequest, 41
 EncodeTransferCallRequest, 41, 42
 GenericCSTAp2, 33
 MakeACSEAssociation, 42
 MakeCall, 42
 MonitorStart, 43
 MonitorStop, 43, 44
 QueryDevice, 44
 ReleaseACSEAssociation, 44
 RetrieveCall, 44
 SessionObject, 46
 SetDoNotDisturb, 44
 SetMessageWaiting, 45
 SnapshotDevice, 45
 ThreadContext, 46
 TransferCall, 45
 Com::Objsys::Csta::Phase2::IETF_CSTAp2, 92
 IETF_CSTAp2, 92
 Com::Objsys::Csta::Phase2::Phase2Opcodes, 111
 Opcodes, 111
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 47
 AlternateCall, 50
 AnswerCall, 50
 ClearConnection, 50
 ClearDoNotDisturb, 51
 ClearMessageWaiting, 51
 ConferenceCall, 51
 ConsultationCall, 51
 EncodeACSEConnectionRequest, 52
 EncodeACSEReleaseRequest, 52
 EncodeAlternateCallRequest, 52
 EncodeAnswerCallRequest, 53
 EncodeClearConnectionRequest, 53
 EncodeConferenceCallRequest, 54
 EncodeConsultationCallRequest, 54
 EncodeGetAgentStateRequest, 54
 EncodeGetDNDRequest, 55
 EncodeGetLogicalDevInfoRequest, 55
 EncodeGetPhysicalDevInfoRequest, 55
 EncodeGetSFDevicesRequest, 56
 EncodeHoldCallRequest, 56
 EncodeMakeCallRequest, 56
 EncodeMonitorStartRequest, 57
 EncodeMonitorStopRequest, 57
 EncodeRetrieveCallRequest, 58
 EncodeROSERequestHeader, 58
 EncodeSendDataRequest, 58
 EncodeSetAgentStateRequest, 59
 EncodeSetDisplayRequest, 59
 EncodeSetOrClearDNDRequest, 60
 EncodeSetOrClearMWRequest, 60
 EncodeSetRingerStatusRequest, 60
 EncodeSingleStepTransferRequest, 61
 EncodeSnapshotCallRequest, 61
 EncodeSnapshotDeviceRequest, 62
 EncodeStartDataPathRequest, 62
 EncodeStopDataPathRequest, 62
 EncodeTransferCallRequest, 63
 GenericCSTAp3, 49
 GetAgentState, 63
 GetDoNotDisturb, 64
 GetLogicalDevInfo, 64
 GetPhysicalDevInfo, 64
 GetSFDevices, 64
 HoldCall, 65
 MakeACSEAssociation, 65
 MakeCall, 65
 MonitorStart, 65
 MonitorStop, 66
 ReleaseACSEAssociation, 66
 RetrieveCall, 67
 RingDevice, 67
 SendData, 67
 SessionObject, 71
 SetAgentState, 67, 68
 SetDisplay, 68
 SetDoNotDisturb, 68
 SetMessageWaiting, 68
 SingleStepTransfer, 69
 SnapshotCall, 69

- SnapshotDevice, 69
- StartDataPath, 70
- StopDataPath, 70
- StopRing, 70
- ThreadContext, 71
- TransferCall, 70, 71
- Com::Objsys::Csta::Phase3::IETF_CSTAp3, 93
 - IETF_CSTAp3, 93
- Com::Objsys::Csta::Phase3::Phase3Opcodes, 112
 - Opcodes, 112
- Com::Objsys::Csta::Phase3::SingleStepTransferInfo, 126
 - TransferFromDevice, 126
 - TransferringCallID, 126
 - TransferToDevice, 126
- Com::Objsys::Csta::Phase3::TransferCallInfo, 129
 - ActiveDeviceCallID, 129
 - ActiveDeviceNumber, 129
 - ConnectedDevice, 129
 - HeldDevice, 129
 - HeldDeviceCallID, 129
 - HeldDeviceNumber, 129
- Com::Objsys::Csta::Xml::GenericXML, 72
 - AnswerCall, 74
 - ClearConnection, 74
 - ClearDoNotDisturb, 75
 - ClearMessageWaiting, 75
 - ConsultationCall, 75
 - EncodeAnswerCall, 75, 76
 - EncodeClearConnection, 76
 - EncodeConsultationCall, 76
 - EncodeGetAgentState, 76
 - EncodeGetDND, 77
 - EncodeGetSFDevices, 77
 - EncodeHoldCall, 77
 - EncodeInvokeID, 77
 - EncodeMakeCall, 78
 - EncodeMonitorStart, 78
 - EncodeMonitorStop, 78
 - EncodeRetrieveCall, 79
 - EncodeSendData, 79
 - EncodeSetAgentState, 79, 80
 - EncodeSetDisplay, 80
 - EncodeSetDND, 80
 - EncodeSetMsgWaiting, 80
 - EncodeSetRingerStatus, 81
 - EncodeSingleStepTransfer, 81
 - EncodeSnapshotCall, 81
 - EncodeSnapshotDevice, 82
 - EncodeStartDataPath, 82
 - EncodeStartSession, 82
 - EncodeStopDataPath, 82
 - EncodeStopSession, 83
 - EncodeTransferCall, 83
 - GenericXML, 74
 - GetAgentState, 83
 - GetDoNotDisturb, 83
 - GetSFDevices, 84
 - HoldCall, 84
 - MakeCall, 84
 - MonitorStart, 84, 85
 - MonitorStop, 85
 - MonitorStopAtDevice, 85
 - RetrieveCall, 85
 - RingDevice, 86
 - SendData, 86
 - SessionObject, 90
 - SetAgentState, 86
 - SetDisplay, 87
 - SetDoNotDisturb, 87
 - SetMessageWaiting, 87
 - SingleStepTransfer, 87
 - SnapshotCall, 88
 - SnapshotDevice, 88
 - StartDataPath, 88
 - StartSession, 88, 89
 - StopDataPath, 89
 - StopRing, 89
 - StopSession, 89
 - ThreadContext, 90
 - TransferCall, 89
- ConferenceCall
 - Com::Objsys::Csta::Phase3::GenericCSTAp3, 51
- ConnectedDevice
 - Com::Objsys::Csta::Phase3::TransferCallInfo, 129
- ConnectionCallback
 - Com::Objsys::Csta::Common::PBXSession, 103
- ConnectionLostCallback
 - Com::Objsys::Csta::Common::PBXSession, 106
- ConsultationCall
 - Com::Objsys::Csta::Phase1::GenericCSTAp1, 20
 - Com::Objsys::Csta::Phase2::GenericCSTAp2, 35
 - Com::Objsys::Csta::Phase3::GenericCSTAp3, 51
 - Com::Objsys::Csta::Xml::GenericXML, 75
- DebugClientCallback
 - Com::Objsys::Csta::Common::PBXSession, 106
- DebugMode
 - Com::Objsys::Csta::Common::PBXSession, 106
- DeviceDataTypes
 - Com::Objsys::Csta::Devices::PanasonicKXTDE, 96
- DivertCall
 - Com::Objsys::Csta::Phase1::GenericCSTAp1, 20
 - Com::Objsys::Csta::Phase2::GenericCSTAp2, 35
- EncodeACSEConnectionRequest
 - Com::Objsys::Csta::Devices::Alcatel4400, 10
 - Com::Objsys::Csta::Devices::AlcatelOXO, 12
 - Com::Objsys::Csta::Devices::PanasonicKXTDE, 97

Com::Objsys::Csta::Devices::SiemensCap, 117
 Com::Objsys::Csta::Devices::SiemensHipath3000p3, 122
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 21
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 36
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 52
 EncodeACSEReleaseRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 21
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 36
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 52
 EncodeAlternateCallRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 52
 EncodeAnswerCall
 Com::Objsys::Csta::Xml::GenericXML, 75, 76
 EncodeAnswerCallRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 21, 22
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 36, 37
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 53
 EncodeClearConnection
 Com::Objsys::Csta::Xml::GenericXML, 76
 EncodeClearConnectionRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 22
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 37
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 53
 EncodeConferenceCallRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 54
 EncodeConsultationCall
 Com::Objsys::Csta::Xml::GenericXML, 76
 EncodeConsultationCallRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 22
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 37
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 54
 EncodeDivertCallRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 23
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 38
 EncodeGetAgentState
 Com::Objsys::Csta::Xml::GenericXML, 76
 EncodeGetAgentStateRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 54
 EncodeGetDND
 Com::Objsys::Csta::Xml::GenericXML, 77
 EncodeGetDNDRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 55
 EncodeGetLogicalDevInfoRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 55
 EncodeGetPhysicalDevInfoRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 55
 EncodeGetSFDevices
 Com::Objsys::Csta::Xml::GenericXML, 77
 EncodeGetSFDevicesRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 56
 EncodeHoldCall
 Com::Objsys::Csta::Xml::GenericXML, 77
 EncodeHoldCallRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 56
 EncodeInvokeID
 Com::Objsys::Csta::Xml::GenericXML, 77
 EncodeMakeCall
 Com::Objsys::Csta::Xml::GenericXML, 78
 EncodeMakeCallRequest
 Com::Objsys::Csta::Devices::Alcatel4400, 10
 Com::Objsys::Csta::Devices::PanasonicKXTDE, 98
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 23
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 38
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 56
 EncodeMonitorStart
 Com::Objsys::Csta::Xml::GenericXML, 78
 EncodeMonitorStartRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 23, 24
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 38, 39
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 57
 EncodeMonitorStop
 Com::Objsys::Csta::Xml::GenericXML, 78
 EncodeMonitorStopRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 24
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 39
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 57
 EncodeQueryDeviceRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 24
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 40
 EncodeRetrieveCall
 Com::Objsys::Csta::Xml::GenericXML, 79
 EncodeRetrieveCallRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 25
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 40
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 58
 EncodeROSERequestHeader
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 25
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 40
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 58
 EncodeSendData
 Com::Objsys::Csta::Xml::GenericXML, 79
 EncodeSendDataRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 58
 EncodeSetAgentState
 Com::Objsys::Csta::Xml::GenericXML, 79, 80
 EncodeSetAgentStateRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 59
 EncodeSetDisplay
 Com::Objsys::Csta::Xml::GenericXML, 80
 EncodeSetDisplayRequest
 Com::Objsys::Csta::Devices::PanasonicKXTDE, 98
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 59
 EncodeSetDND

Com::Objsys::Csta::Xml::GenericXML, 80
 EncodeSetMsgWaiting
 Com::Objsys::Csta::Xml::GenericXML, 80
 EncodeSetOrClearDNDRRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 25
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 40
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 60
 EncodeSetOrClearMWRRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 26
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 41
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 60
 EncodeSetRingerStatus
 Com::Objsys::Csta::Xml::GenericXML, 81
 EncodeSetRingerStatusRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 60
 EncodeSingleStepTransfer
 Com::Objsys::Csta::Xml::GenericXML, 81
 EncodeSingleStepTransferRequest
 Com::Objsys::Csta::Devices::SiemensHipath3000p3, 123
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 61
 EncodeSnapshotCall
 Com::Objsys::Csta::Xml::GenericXML, 81
 EncodeSnapshotCallRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 61
 EncodeSnapshotDevice
 Com::Objsys::Csta::Xml::GenericXML, 82
 EncodeSnapshotDeviceRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 26
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 41
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 62
 EncodeStartDataPath
 Com::Objsys::Csta::Xml::GenericXML, 82
 EncodeStartDataPathRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 62
 EncodeStartSession
 Com::Objsys::Csta::Xml::GenericXML, 82
 EncodeStopDataPath
 Com::Objsys::Csta::Xml::GenericXML, 82
 EncodeStopDataPathRequest
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 62
 EncodeStopSession
 Com::Objsys::Csta::Xml::GenericXML, 83
 EncodeTransferCall
 Com::Objsys::Csta::Xml::GenericXML, 83
 EncodeTransferCallRequest
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 26, 27
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 41, 42
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 63
 ExceptionCallback
 Com::Objsys::Csta::Common::PBXSession, 106
 GenericCSTAp1
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 18
 GenericCSTAp2
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 33
 GenericCSTAp3
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 49
 GenericXML
 Com::Objsys::Csta::Xml::GenericXML, 74
 GetAgentState
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 63
 Com::Objsys::Csta::Xml::GenericXML, 83
 GetDeviceData
 Com::Objsys::Csta::Devices::PanasonicKXTDE, 98
 GetDoNotDisturb
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 64
 Com::Objsys::Csta::Xml::GenericXML, 83
 GetGroupMembers
 Com::Objsys::Csta::Devices::PanasonicKXTDE, 99
 GetLogicalDevInfo
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 64
 GetPhysicalDevInfo
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 64
 GetSFDevices
 Com::Objsys::Csta::Devices::PanasonicKXTDE, 99
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 64
 Com::Objsys::Csta::Xml::GenericXML, 84
 HeldDevice
 Com::Objsys::Csta::Phase3::TransferCallInfo, 129
 HeldDeviceCallID
 Com::Objsys::Csta::Phase3::TransferCallInfo, 129
 HeldDeviceNumber
 Com::Objsys::Csta::Phase3::TransferCallInfo, 129
 HoldCall
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 65
 Com::Objsys::Csta::Xml::GenericXML, 84
 IETF_CSTAp1
 Com::Objsys::Csta::Phase1::IETF_CSTAp1, 91
 IETF_CSTAp2
 Com::Objsys::Csta::Phase2::IETF_CSTAp2, 92
 IETF_CSTAp3
 Com::Objsys::Csta::Phase3::IETF_CSTAp3, 93
 LoggingFolder
 Com::Objsys::Csta::Common::PBXSessionHelper, 109
 MakeACSEAssociation
 Com::Objsys::Csta::Devices::Alcatel4400, 10
 Com::Objsys::Csta::Devices::AlcatelOXO, 13
 Com::Objsys::Csta::Devices::PhilipsSopho, 113
 Com::Objsys::Csta::Devices::SiemensCap, 118
 Com::Objsys::Csta::Devices::SiemensHipath3000p3, 123

Com::Objsys::Csta::Phase1::GenericCSTAp1, 27
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 42
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 65
MakeCall
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 27
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 42
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 65
 Com::Objsys::Csta::Xml::GenericXML, 84
MaxReceiveTimeout
 Com::Objsys::Csta::Common::PBXSession, 106
MessageEncoding
 Com::Objsys::Csta::Common::PBXSession, 106
MonitorStart
 Com::Objsys::Csta::Devices::VodiaSNOMOne, 132
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 27, 28
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 43
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 65
 Com::Objsys::Csta::Xml::GenericXML, 84, 85
MonitorStop
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 28
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 43, 44
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 66
 Com::Objsys::Csta::Xml::GenericXML, 85
MonitorStopAtDevice
 Com::Objsys::Csta::Xml::GenericXML, 85
Opcodes
 Com::Objsys::Csta::Phase1::Phase1Opcodes, 110
 Com::Objsys::Csta::Phase2::Phase2Opcodes, 111
 Com::Objsys::Csta::Phase3::Phase3Opcodes, 112
Open
 Com::Objsys::Csta::Common::PBXSession, 103
PanasonicKXTDA
 Com::Objsys::Csta::Devices::PanasonicKXTDA, 95
PanasonicKXTDE
 Com::Objsys::Csta::Devices::PanasonicKXTDE, 97
PanasonicNCP
 Com::Objsys::Csta::Devices::PanasonicNCP, 101
PBXSession
 Com::Objsys::Csta::Common::PBXSession, 103
PBXSystem
 Com::Objsys::Csta::Common::PBXSession, 106
PDFStart
 Com::Objsys::Csta::Devices::PanasonicKXTDE, 99
PDFStop
 Com::Objsys::Csta::Devices::PanasonicKXTDE, 99
PhilipsSopho
 Com::Objsys::Csta::Devices::PhilipsSopho, 113
Port
 Com::Objsys::Csta::Common::PBXSession, 107
QueryDevice
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 29
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 44
ReadBuffer
 Com::Objsys::Csta::Common::SocketState, 127
ReadBuffers
 Com::Objsys::Csta::Common::SocketState, 127
ReleaseACSEAssociation
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 29
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 44
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 66
ReleaseControlRight
 Com::Objsys::Csta::Devices::PanasonicKXTDE, 99
ResetDisplay
 Com::Objsys::Csta::Devices::PanasonicKXTDE, 100
ResponseFromPBX
 Com::Objsys::Csta::Common::CSTAContext, 14
 Com::Objsys::Csta::Common::CSTAResponseInfo, 15
ResponsesFromPBX
 Com::Objsys::Csta::Common::CSTAContext, 14
 Com::Objsys::Csta::Common::CSTAResponseInfo, 15
RetrieveCall
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 29
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 44
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 67
 Com::Objsys::Csta::Xml::GenericXML, 85
RingDevice
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 67
 Com::Objsys::Csta::Xml::GenericXML, 86
SendACSEMessage
 Com::Objsys::Csta::Common::PBXSession, 104
SendData
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 67
 Com::Objsys::Csta::Xml::GenericXML, 86
SendKmeMessage
 Com::Objsys::Csta::Devices::PanasonicKXTDE, 100
SendMessage
 Com::Objsys::Csta::Common::PBXSession, 104
SendXMLMessage
 Com::Objsys::Csta::Common::PBXSession, 104, 105
SendXMLSession
 Com::Objsys::Csta::Common::PBXSession, 105
SessionObject
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 31
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 46
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 71
 Com::Objsys::Csta::Xml::GenericXML, 90
SetAgentState

Com::Objsys::Csta::Phase3::GenericCSTAp3, 67,
 68
 Com::Objsys::Csta::Xml::GenericXML, 86
 SetDisplay
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 68
 Com::Objsys::Csta::Xml::GenericXML, 87
 SetDoNotDisturb
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 29
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 44
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 68
 Com::Objsys::Csta::Xml::GenericXML, 87
 SetMessageWaiting
 Com::Objsys::Csta::Devices::PanasonicKXTDE,
 100
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 29
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 45
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 68
 Com::Objsys::Csta::Xml::GenericXML, 87
 SiemensCap
 Com::Objsys::Csta::Devices::SiemensCap, 117
 SiemensHicom300
 Com::Objsys::Csta::Devices::SiemensHicom300,
 120
 SiemensHipath3000p2
 Com::Objsys::Csta::Devices::SiemensHipath3000p2,
 121
 SiemensHipath3000p3
 Com::Objsys::Csta::Devices::SiemensHipath3000p3,
 122
 SiemensHipath4000
 Com::Objsys::Csta::Devices::SiemensHipath4000,
 124
 SiemensRealitis
 Com::Objsys::Csta::Devices::SiemensRealitis, 125
 SingleStepTransfer
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 69
 Com::Objsys::Csta::Xml::GenericXML, 87
 SnapshotCall
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 69
 Com::Objsys::Csta::Xml::GenericXML, 88
 SnapshotDevice
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 30
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 45
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 69
 Com::Objsys::Csta::Xml::GenericXML, 88
 StartDataPath
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 70
 Com::Objsys::Csta::Xml::GenericXML, 88
 StartSession
 Com::Objsys::Csta::Devices::VodiaSNOMOne,
 132, 133
 Com::Objsys::Csta::Xml::GenericXML, 88, 89
 StatusCode
 Com::Objsys::Csta::Common::CSTAResponseInfo,
 15
 Com::Objsys::Csta::Common::CSTAResponseInfo,
 15
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 70
 Com::Objsys::Csta::Xml::GenericXML, 89
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 70
 Com::Objsys::Csta::Xml::GenericXML, 89
 Com::Objsys::Csta::Xml::GenericXML, 89
 Com::Objsys::Csta::Common::PBXSession, 107
 TadiranCoral
 Com::Objsys::Csta::Devices::TadiranCoral, 128
 ThreadContext
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 31
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 46
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 71
 Com::Objsys::Csta::Xml::GenericXML, 90
 TotalLength
 Com::Objsys::Csta::Common::SocketState, 127
 TransferCall
 Com::Objsys::Csta::Phase1::GenericCSTAp1, 30
 Com::Objsys::Csta::Phase2::GenericCSTAp2, 45
 Com::Objsys::Csta::Phase3::GenericCSTAp3, 70,
 71
 Com::Objsys::Csta::Xml::GenericXML, 89
 TransferFromDevice
 Com::Objsys::Csta::Phase3::SingleStepTransferInfo,
 126
 TransferringCallID
 Com::Objsys::Csta::Phase3::SingleStepTransferInfo,
 126
 TransferToDevice
 Com::Objsys::Csta::Phase3::SingleStepTransferInfo,
 126
 UnifyOpenscapeVoice
 Com::Objsys::Csta::Devices::UnifyOpenscapeVoice,
 130
 UnifyOpenscapeX5
 Com::Objsys::Csta::Devices::UnifyOpenscapeX5,
 131
 VodiaSNOMOne
 Com::Objsys::Csta::Devices::VodiaSNOMOne, 132
 WaitForROSEResponse
 Com::Objsys::Csta::Common::PBXSession, 105
 WaitForXMLResponse
 Com::Objsys::Csta::Common::PBXSession, 105

XMLAsyncCallback
Com::Objsys::Csta::Common::PBXSession, [105](#)

XMLClientCallback
Com::Objsys::Csta::Common::PBXSession, [107](#)

XMLResponseFromPBX
Com::Objsys::Csta::Common::CSTAContext, [14](#)
Com::Objsys::Csta::Common::CSTAResponseInfo,
[15](#)

XMLResponsesFromPBX
Com::Objsys::Csta::Common::CSTAContext, [14](#)
Com::Objsys::Csta::Common::CSTAResponseInfo,
[16](#)

XMLSystemStatusCallback
Com::Objsys::Csta::Common::PBXSession, [107](#)