



OpenScape Contact Center Extensions V2R1 Integration Connector

Programming Guide

A31003-S2220-R100-4-7620

Our Quality and Environmental Management Systems are implemented according to the requirements of the ISO9001 and ISO14001 standards and are certified by an external certification company.

Copyright©Unify Software and Solutions GmbH & Co. KG 2014
Mies-van-der-Rohe-Str. 6, 80807 Munich/Germany

All rights reserved.

Reference No.: A31003-S2220-R100-4-7620

The information provided in this document contains merely general descriptions or characteristics of performance which in case of actual use do not always apply as described or which may change as a result of further development of the products.

An obligation to provide the respective characteristics shall only exist if expressly agreed in the terms of contract.

Availability and technical specifications are subject to change without notice.

Unify, OpenScape, OpenStage and HiPath are registered trademarks of Unify Software and Solutions GmbH & Co. KG. All other company, brand, product and service names are trademarks or registered trademarks trademarks or registered trademarks of their respective holders.

Content

1	About this guide	17
1.1	Important product information.....	17
1.2	Overview of contents	17
1.3	Typographical styles.....	18
2	CTI.....	19
2.1	Basics	19
2.1.1	System architecture.....	19
2.1.2	Application development	20
2.1.2.1	Registration of the interface library	20
2.1.2.2	Application examples	20
2.1.2.3	Synchronous and asynchronous method execution	21
2.1.2.4	Troubleshooting.....	22
2.1.2.5	Status changes among extensions and agents	22
2.2	Enumeration types	23
2.2.1	AcdContactTypeEnum	23
2.2.2	AcdDequeueReasonEnum.....	23
2.2.3	AcdFunctionTypeEnum	23
2.2.4	AcdHandlingStateEnum	23
2.2.5	AcdMediaTypeEnum	23
2.2.6	AcdPartyTypeEnum	23
2.2.7	AgentStateOsvHuntgroupEnum	23
2.2.8	ApplicationIdentifierEnum.....	23
2.2.9	AutoAnswerFlagEnum.....	23
2.2.10	BusyStateDetailEnum	23
2.2.11	BusyStateEnum.....	23
2.2.12	CallDirectionEnum.....	23
2.2.13	CallListChangeNotificationEnum.....	23
2.2.14	CallStateEnum	23
2.2.15	ConferenceEventTypeEnum	24
2.2.16	CstaVersionEnum	24
2.2.17	DeviceTypeEnum	24
2.2.18	DoNotDisturbFlagEnum	24
2.2.19	ErrorEnum	24
2.2.20	EventCauseEnum	24
2.2.21	EventTypeEnum.....	24
2.2.22	ForwardingStateEnum.....	24
2.2.23	IPVersionEnum	24
2.2.24	LocalConnectionStateEnum.....	24
2.2.25	LogLevelEnum	24
2.2.26	LogMessageDelimiterEnum	24

Content

2.2.27	MonitorDeviceHandlingFlagEnum	24
2.2.28	MonitorStateEnum	24
2.2.29	NodeTypeEnum	24
2.2.30	ServerConnectionModeEnum	24
2.2.31	ServerConnectionStateEnum	25
2.2.32	ServerTypeEnum	25
2.2.33	SetForwardingActionEnum	25
2.2.34	SiteStateEnum	25
2.2.35	SiteTypeEnum	25
2.2.36	UserAgentStateEnum	25
2.3	"CocCtiConnector" class	26
2.3.1	Properties	26
2.3.1.1	AvailableLicences	26
2.3.1.2	ConfigData	26
2.3.1.3	ServerConnectionMode	26
2.3.1.4	ServerConnectionState	26
2.3.2	Methods	26
2.3.2.1	AcdAvailable	26
2.3.2.2	AcdCheckPassword	27
2.3.2.3	AcdExitPostProcessingState	27
2.3.2.4	AcdGetBusinessUnitForExtension	28
2.3.2.5	AcdGetBusinessUnitForUserId	28
2.3.2.6	AcdGetBusinessUnits	28
2.3.2.7	AcdGetCallInformation	29
2.3.2.8	AcdGetPostProcessingReasons	29
2.3.2.9	AcdGetPostProcessingState	30
2.3.2.10	AcdGetQueues	30
2.3.2.11	AcdGetTeams	31
2.3.2.12	AcdGetUnavailableReasons	31
2.3.2.13	AcdGetUserIdForUserName	31
2.3.2.14	AcdGetUsers	32
2.3.2.15	AcdGetUserState	32
2.3.2.16	AcdGetWorkReasons	33
2.3.2.17	AcdGetWrapupReasons	33
2.3.2.18	AcdLogoff	33
2.3.2.19	AcdLogon	34
2.3.2.20	AcdRegisterForFunctions	34
2.3.2.21	AcdRegisterForGlobalEvents	35
2.3.2.22	AcdRegisterForNodeEvents	35
2.3.2.23	AcdSaveWrapupData	35
2.3.2.24	AcdSetPostProcessingState	36
2.3.2.25	AcdSetPrimaryContact	37
2.3.2.26	AcdUnavailable	37
2.3.2.27	AcdUnregisterForFunctions	38
2.3.2.28	AcdUnregisterForGlobalEvents	38
2.3.2.29	AcdUnregisterForNodeEvents	38
2.3.2.30	AcdWork	39
2.3.2.31	AgentAvailable	39
2.3.2.32	AgentLogoff	40

2.3.2.33	AgentLogon	40
2.3.2.34	AgentUnavailable	41
2.3.2.35	AgentWork	41
2.3.2.36	AlternateCall	41
2.3.2.37	AnswerCall	42
2.3.2.38	CheckMonitoringDevice	42
2.3.2.39	ClearConnection	43
2.3.2.40	Close	43
2.3.2.41	ConferenceCall	43
2.3.2.42	ConsultCall	44
2.3.2.43	CreateCallback	44
2.3.2.44	DeflectCall	45
2.3.2.45	DeleteCallback	45
2.3.2.46	DialDigits	45
2.3.2.47	GetAgentState	46
2.3.2.48	GetAgentStateOsvHuntgroup	46
2.3.2.49	GetDoNotDisturb	46
2.3.2.50	GetErrorAsString	47
2.3.2.51	GetExtensionCallList	47
2.3.2.52	GetForwarding	47
2.3.2.53	GetPhoneState	47
2.3.2.54	HoldCall	48
2.3.2.55	IsExtensionOutOfService	48
2.3.2.56	MakeCall	48
2.3.2.57	Open	49
2.3.2.58	ReconnectCall	49
2.3.2.59	RetrieveCall	50
2.3.2.60	SendDTMF	50
2.3.2.61	SetAgentStateOsvHuntgroup	50
2.3.2.62	SetForwarding	51
2.3.2.63	SingleStepTransferCall	51
2.3.2.64	SnapshotDevice	52
2.3.2.65	SnapshotDeviceAdvanced	52
2.3.2.66	StartMonitorDevice	52
2.3.2.67	StopMonitorDevice	53
2.3.2.68	TransferCall	53
2.3.3	Events	54
2.3.3.1	OnAcdAgentStateChangeInfoEvent	54
2.3.3.2	OnAcdAvailableEvent	54
2.3.3.3	OnAcdCheckPasswordEvent	54
2.3.3.4	OnAcdContactStateChangeInfoEvent	54
2.3.3.5	OnAcdContactStateInfoEvent	54
2.3.3.6	OnAcdExitPostProcessingStateEvent	55
2.3.3.7	OnAcdGetBusinessUnitForExtensionEvent	55
2.3.3.8	OnAcdGetBusinessUnitForUserIdEvent	55
2.3.3.9	OnAcdGetBusinessUnitsEvent	55
2.3.3.10	OnAcdGetCallInformationEvent	55
2.3.3.11	OnAcdGetPostProcessingReasonsEvent	56
2.3.3.12	OnAcdGetPostProcessingStateEvent	56
2.3.3.13	OnAcdGetQueuesEvent	56
2.3.3.14	OnAcdGetTeamsEvent	56
2.3.3.15	OnAcdGetUnavailableReasonsEvent	56

Content

2.3.3.16	OnAcdGetUserIdForUserNameEvent	57
2.3.3.17	OnAcdGetUsersEvent	57
2.3.3.18	OnAcdGetUserStateEvent	57
2.3.3.19	OnAcdGetWorkReasonsEvent	57
2.3.3.20	OnAcdGetWrapupReasonsEvent	57
2.3.3.21	OnAcdLogoffEvent	58
2.3.3.22	OnAcdLogonEvent	58
2.3.3.23	OnAcdRegisterForFunctionsEvent	58
2.3.3.24	OnAcdRegisterForGlobalEventsEvent	58
2.3.3.25	OnAcdRegisterForNodeEventsEvent	58
2.3.3.26	OnAcdSaveWrapupDataEvent	59
2.3.3.27	OnAcdSetPostProcessingStateEvent	59
2.3.3.28	OnAcdSetPrimaryContactEvent	59
2.3.3.29	OnAcdUnavailableEvent	59
2.3.3.30	OnAcdUnregisterForFunctionsEvent	59
2.3.3.31	OnAcdUnregisterForGlobalEventsEvent	60
2.3.3.32	OnAcdUnregisterForNodeEventsEvent	60
2.3.3.33	OnAcdUserStateChangeInfoEvent	60
2.3.3.34	OnAcdVoiceRingingInfoEvent	60
2.3.3.35	OnAcdWorkEvent	60
2.3.3.36	OnAgentAvailableEvent	61
2.3.3.37	OnAgentLogoffEvent	61
2.3.3.38	OnAgentLogonEvent	61
2.3.3.39	OnAgentStateChangeInfoEvent	61
2.3.3.40	OnAgentUnavailableEvent	61
2.3.3.41	OnAgentWorkEvent	62
2.3.3.42	OnAlternateCallEvent	62
2.3.3.43	OnAnswerCallEvent	62
2.3.3.44	OnCallbackInfoEvent	62
2.3.3.45	OnCallDisconnectOnHeldInfoEvent	62
2.3.3.46	OnCallEstablishedInfoEvent	63
2.3.3.47	OnCallPickupInfoEvent	63
2.3.3.48	OnCheckMonitoringDeviceEvent	63
2.3.3.49	OnClearConnectionEvent	63
2.3.3.50	OnConferenceCallEvent	63
2.3.3.51	OnConferenceInfoEvent	64
2.3.3.52	OnConsultCallEvent	64
2.3.3.53	OnCreateCallbackEvent	64
2.3.3.54	OnDeflectCallEvent	64
2.3.3.55	OnDeleteCallbackEvent	64
2.3.3.56	OnDialDigitsEvent	65
2.3.3.57	OnErrorEvent	65
2.3.3.58	OnFailoverInfoEvent	65
2.3.3.59	OnGetAgentStateEvent	65
2.3.3.60	OnGetAgentStateOsvHuntgroupEvent	65
2.3.3.61	OnGetDoNotDisturbEvent	66
2.3.3.62	OnGetForwardingEvent	66
2.3.3.63	OnHoldCallEvent	66
2.3.3.64	OnLoggingEvent	66
2.3.3.65	OnMakeCallEvent	66
2.3.3.66	OnPhoneCallStateChangeInfoEvent	67
2.3.3.67	OnPhoneClearedInfoEvent	67

2.3.3.68	OnPhoneDialingInfoEvent	67
2.3.3.69	OnPhoneRingingInfoEvent	67
2.3.3.70	OnPhoneStateBusyInfoEvent	67
2.3.3.71	OnPhoneStateChangeInfoEvent	67
2.3.3.72	OnReconnectCallEvent	68
2.3.3.73	OnRetrieveCallEvent	68
2.3.3.74	OnSendDTMFEvent	68
2.3.3.75	OnServerConnectionStateChangeEvent	68
2.3.3.76	OnSetAgentStateOsvHuntgroupEvent	68
2.3.3.77	OnSetForwardingEvent	69
2.3.3.78	OnSingleStepTransferCallEvent	69
2.3.3.79	OnSiteStateInfoEvent	69
2.3.3.80	OnSnapshotDeviceAdvancedEvent	69
2.3.3.81	OnSnapshotDeviceEvent	69
2.3.3.82	OnStartMonitorDeviceEvent	70
2.3.3.83	OnStopMonitorDeviceEvent	70
2.3.3.84	OnTransferCallEvent	70
2.3.3.85	OnTransferCompletedInfoEvent	70
2.3.3.86	OnTransferredInfoEvent	70
2.4	Other classes	71
2.4.1	AcdContactDataListParamX	71
2.4.1.1	Properties	71
2.4.1.2	Methods	71
2.4.2	AcdFunctionTypeListParamX	71
2.4.2.1	Properties	71
2.4.2.2	Methods	71
2.4.3	AcdWrapupReasonKeyDataListParamX	72
2.4.3.1	Properties	72
2.4.3.2	Methods	72
2.5	Interfaces	73
2.5.1	IAcdAgentStateChangeInfoDataX	73
2.5.1.1	Properties	73
2.5.2	IAcdAvailableDataX	74
2.5.2.1	Properties	74
2.5.3	IAcdBusinessUnitsDataX	75
2.5.3.1	Properties	75
2.5.4	IAcdBusinessUnitsListX	75
2.5.4.1	Properties	75
2.5.4.2	Methods	75
2.5.5	IAcdCheckPasswordDataX	76
2.5.5.1	Properties	76
2.5.6	IAcdContactDataListX	77
2.5.6.1	Properties	77
2.5.6.2	Methods	77
2.5.7	IAcdContactDataX	77
2.5.7.1	Properties	77
2.5.8	IAcdContactStateAdvancedDataListX	78
2.5.8.1	Properties	78
2.5.8.2	Methods	78

Content

2.5.9	IAcdContactStateAdvancedDataX	78
2.5.9.1	Properties	78
2.5.10	IAcdContactStateChangeInfoDataX	80
2.5.10.1	Properties	80
2.5.11	IAcdContactStateDataListX	81
2.5.11.1	Properties	81
2.5.11.2	Methods	81
2.5.12	IAcdContactStateDataX	82
2.5.12.1	Properties	82
2.5.13	IAcdContactStateInfoDataX	83
2.5.13.1	Properties	83
2.5.14	IAcdExitPostProcessingStateDataX	85
2.5.14.1	Properties	85
2.5.15	IAcdGetBusinessUnitForExtensionDataX	86
2.5.15.1	Properties	86
2.5.16	IAcdGetBusinessUnitForUserIdDataX	87
2.5.16.1	Properties	87
2.5.17	IAcdGetBusinessUnitsDataX	88
2.5.17.1	Properties	88
2.5.18	IAcdGetCallInformationDataX	89
2.5.18.1	Properties	89
2.5.18.2	Methods	90
2.5.19	IAcdGetPostProcessingReasonsDataX	91
2.5.19.1	Properties	91
2.5.20	IAcdGetPostProcessingStateDataX	91
2.5.20.1	Properties	91
2.5.21	IAcdGetQueuesDataX	92
2.5.21.1	Properties	92
2.5.21.2	Methods	93
2.5.22	IAcdGetTeamsDataX	93
2.5.22.1	Properties	93
2.5.22.2	Methods	94
2.5.23	IAcdGetUnavailableReasonsDataX	94
2.5.23.1	Properties	94
2.5.23.2	Methods	95
2.5.24	IAcdGetUserIdForUserNameDataX	95
2.5.24.1	Properties	95
2.5.25	IAcdGetUsersDataX	96
2.5.25.1	Properties	96
2.5.26	IAcdGetUserStateDataX	97
2.5.26.1	Properties	97
2.5.27	IAcdGetWorkReasonsDataX	98
2.5.27.1	Properties	99
2.5.28	IAcdGetWrapupReasonsDataX	99
2.5.28.1	Properties	99
2.5.28.2	Methods	100
2.5.29	IAcdLogoffDataX	100
2.5.29.1	Properties	100

2.5.30	IAcdLogonDataX	101
2.5.30.1	Properties	101
2.5.31	IAcdMediaTypeDataX	102
2.5.31.1	Properties	102
2.5.32	IAcdMediaTypeListX	103
2.5.32.1	Properties	103
2.5.32.2	Methods	103
2.5.33	IAcdPostProcessingReasonsDataX	103
2.5.33.1	Properties	103
2.5.34	IAcdPostProcessingReasonsListX	104
2.5.34.1	Properties	104
2.5.34.2	Methods	104
2.5.35	IAcdPostProcessingServPermFlagsX	105
2.5.35.1	Properties	105
2.5.35.2	Methods	105
2.5.36	IAcdQueueDataListX	105
2.5.36.1	Properties	105
2.5.36.2	Methods	105
2.5.37	IAcdQueueDataX	106
2.5.37.1	Properties	106
2.5.38	IAcdRegisterForFunctionsDataX	107
2.5.38.1	Properties	107
2.5.39	IAcdRegisterForGlobalEventsDataX	108
2.5.39.1	Properties	108
2.5.40	IAcdRegisterForNodeEventsDataX	108
2.5.40.1	Properties	108
2.5.41	IAcdSaveWrapupDataX	109
2.5.41.1	Properties	110
2.5.41.2	Methods	110
2.5.42	IAcdSetPostProcessingStateDataX	111
2.5.42.1	Properties	111
2.5.42.2	Methods	111
2.5.43	IAcdSetPrimaryContactDataX	112
2.5.43.1	Properties	112
2.5.43.2	Methods	112
2.5.44	IAcdTeamDataListX	112
2.5.44.1	Properties	113
2.5.44.2	Methods	113
2.5.45	IAcdTeamDataX	113
2.5.45.1	Properties	113
2.5.46	IAcdUnavailableDataX	114
2.5.46.1	Properties	114
2.5.47	IAcdUnavailableReasonDataListX	115
2.5.47.1	Properties	115
2.5.47.2	Methods	115
2.5.48	IAcdUnavailableReasonDataX	115
2.5.48.1	Properties	115
2.5.49	IAcdUnregisterForFunctionsDataX	116

Content

2.5.49.1	Properties	116
2.5.50	IAcdUnregisterForGlobalEventsDataX.....	117
2.5.50.1	Properties	117
2.5.51	IAcdUnregisterForNodeEventsDataX	118
2.5.51.1	Properties	118
2.5.52	IAcdUserConfigDataX	119
2.5.52.1	Properties	119
2.5.53	IAcdUserDataX.....	119
2.5.53.1	Properties	119
2.5.54	IAcdUserDataListX	121
2.5.54.1	Properties	121
2.5.54.2	Methods.....	121
2.5.55	IAcdUserStateChangeInfoDataX	121
2.5.55.1	Properties	121
2.5.56	IAcdVoiceRingingInfoDataX.....	122
2.5.56.1	Properties	122
2.5.57	IAcdWorkDataX.....	126
2.5.57.1	Properties	126
2.5.58	IAcdWorkReasonDataListX.....	127
2.5.58.1	Properties	127
2.5.58.2	Methods.....	127
2.5.59	IAcdWorkReasonDataX	127
2.5.59.1	Properties	127
2.5.60	IAcdWrapupReasonDataX	128
2.5.60.1	Properties	128
2.5.61	IAcdWrapupReasonKeyDataListX	129
2.5.61.1	Properties	129
2.5.61.2	Methods.....	129
2.5.62	IAcdWrapupReasonKeyDataX.....	129
2.5.62.1	Properties	129
2.5.63	IAcdWrapupReasonsDataListX.....	130
2.5.63.1	Properties	130
2.5.63.2	Methods.....	130
2.5.64	IAgentAvailableDataX	130
2.5.64.1	Properties	130
2.5.64.2	Methods.....	131
2.5.65	IAgentLogoffDataX.....	131
2.5.65.1	Properties	131
2.5.65.2	Methods.....	133
2.5.66	IAgentLogonDataX	133
2.5.66.1	Properties	133
2.5.66.2	Methods.....	134
2.5.67	IAgentStateChangeInfoDataX.....	134
2.5.67.1	Properties	134
2.5.68	IAgentUnavailableDataX	135
2.5.68.1	Properties	135
2.5.68.2	Methods.....	136
2.5.69	IAgentWorkDataX.....	136

2.5.69.1	Properties	136
2.5.69.2	Methods	137
2.5.70	IAlternateCallDataX	138
2.5.70.1	Properties	138
2.5.70.2	Methods	138
2.5.71	IAnswerCallDataX	139
2.5.71.1	Properties	139
2.5.71.2	Methods	139
2.5.72	ICallbackInfoDataX	140
2.5.72.1	Properties	140
2.5.73	ICallDisconnectOnHeldInfoDataX	140
2.5.73.1	Properties	140
2.5.74	ICallEstablishedInfoDataX	141
2.5.74.1	Properties	141
2.5.75	ICallPickupInfoDataX	142
2.5.75.1	Properties	142
2.5.76	ICheckMonitoringDeviceDataX	143
2.5.76.1	Properties	143
2.5.76.2	Methods	144
2.5.77	IClearConnectionDataX	144
2.5.77.1	Properties	144
2.5.77.2	Methods	145
2.5.78	ICommonConfigDataX	145
2.5.78.1	Properties	145
2.5.79	IConferenceCallDataX	146
2.5.79.1	Properties	146
2.5.79.2	Methods	146
2.5.80	IConferenceInfoDataX	147
2.5.80.1	Properties	147
2.5.81	IConfigDataX	149
2.5.81.1	Properties	149
2.5.82	IConnectionDataX	149
2.5.82.1	Properties	149
2.5.83	IConnectionListX	150
2.5.83.1	Properties	150
2.5.83.2	Methods	151
2.5.84	IConsultCallDataX	151
2.5.84.1	Properties	151
2.5.84.2	Methods	152
2.5.85	ICreateCallbackDataX	152
2.5.85.1	Properties	152
2.5.85.2	Methods	153
2.5.86	ICstaErrorX	153
2.5.86.1	Properties	153
2.5.86.2	Methods	154
2.5.87	ICtiConnectivityConfigDataX	154
2.5.87.1	Properties	154
2.5.88	IDeflectCallDataX	155

Content

2.5.88.1	Properties	155
2.5.88.2	Methods	156
2.5.89	IDeleteCallbackDataX	156
2.5.89.1	Properties	156
2.5.89.2	Methods	157
2.5.90	IDialDigitsDataX	157
2.5.90.1	Properties	157
2.5.90.2	Methods	158
2.5.91	IExtensionCallDataX	158
2.5.91.1	Properties	158
2.5.92	IExtensionCallListX	159
2.5.92.1	Properties	160
2.5.92.2	Methods	160
2.5.93	IFailoverInfoDataX	160
2.5.93.1	Properties	160
2.5.94	IGetAgentStateDataX	161
2.5.94.1	Properties	161
2.5.94.2	Methods	162
2.5.95	IGetAgentStateOsvHuntgroupDataX	162
2.5.95.1	Properties	162
2.5.95.2	Methods	163
2.5.96	IGetDoNotDisturbDataX	163
2.5.96.1	Properties	163
2.5.96.2	Methods	164
2.5.97	IGetExtensionCallListDataX	164
2.5.97.1	Properties	164
2.5.98	IGetForwardingDataX	165
2.5.98.1	Properties	165
2.5.98.2	Methods	166
2.5.99	IGetPhoneStateDataX	166
2.5.99.1	Properties	166
2.5.100	IHoldCallDataX	167
2.5.100.1	Properties	167
2.5.100.2	Methods	167
2.5.101	ILoggingConfigDataX	168
2.5.101.1	Properties	168
2.5.102	IMakeCallDataX	170
2.5.102.1	Properties	170
2.5.102.2	Methods	170
2.5.103	IOldCallIdListX	171
2.5.103.1	Properties	171
2.5.103.2	Methods	171
2.5.104	IPartyDataX	171
2.5.104.1	Properties	171
2.5.105	IPartyListX	172
2.5.105.1	Properties	172
2.5.105.2	Methods	172
2.5.106	IPhoneCallStateChangeInfoDataX	173

2.5.106.1 Properties	173
2.5.107 IPhoneClearedInfoDataX	174
2.5.107.1 Properties	174
2.5.108 IPhoneDialingInfoDataX	174
2.5.108.1 Properties	175
2.5.109 IPhoneRingingInfoDataX	175
2.5.109.1 Properties	175
2.5.110 IPhoneStateBusyInfoDataX	176
2.5.110.1 Properties	177
2.5.111 IPhoneStateChangeInfoDataX	178
2.5.111.1 Properties	178
2.5.112 IReconnectCallDataX	179
2.5.112.1 Properties	179
2.5.112.2 Methods	179
2.5.113 IRetrieveCallDataX	180
2.5.113.1 Properties	180
2.5.113.2 Methods	180
2.5.114 ISendDTMFDataX	181
2.5.114.1 Properties	181
2.5.114.2 Methods	181
2.5.115 IServerConnectionStateChangeDataX	182
2.5.115.1 Properties	182
2.5.116 ISetAgentStateOsvHuntgroupDataX	182
2.5.116.1 Properties	182
2.5.116.2 Methods	183
2.5.116.3 ISetForwardingDataX	183
2.5.116.4 Properties	183
2.5.116.5 Methods	184
2.5.117 ISingleStepTransferCallDataX	184
2.5.117.1 Properties	184
2.5.117.2 Methods	185
2.5.118 ISiteStateInfoDataX	185
2.5.118.1 Properties	185
2.5.119 ISnapshotAdvCallListX	187
2.5.119.1 Properties	187
2.5.119.2 Methods	187
2.5.120 ISnapshotAdvCallX	187
2.5.120.1 Properties	187
2.5.120.2 Methods	188
2.5.121 ISnapshotDeviceAdvancedDataX	188
2.5.121.1 Properties	188
2.5.121.2 Methods	189
2.5.122 ISnapshotDeviceDataX	189
2.5.122.1 Properties	189
2.5.123 Methods	190
2.5.124 IStartMonitorDeviceDataX	191
2.5.124.1 Properties	191
2.5.124.2 Methods	192

Content

2.5.125	IStopMonitorDeviceDataX	193
2.5.125.1	Properties	193
2.5.125.2	Methods	194
2.5.126	ITransferCallDataX	194
2.5.126.1	Properties	195
2.5.126.2	Methods	195
2.5.127	ITransferCompletedInfoDataX	196
2.5.127.1	Properties	196
2.5.128	ITransferredInfoDataX	198
2.5.128.1	Properties	198
2.5.129	IUserAgentServPermFlagsX	199
2.5.129.1	Properties	199
3	Realtime	201
3.1	Basics	201
3.1.1	System architecture	201
3.1.2	Application example COM DLL	202
3.1.2.1	Registration of the interface library	202
3.1.2.2	Establishing a connection with the Connectivity Server	202
3.1.2.3	Receiving data	204
3.1.2.4	Agents	204
3.1.3	Application development with Java API	205
3.1.3.1	Integration of the Java API	205
3.1.3.2	Establishing a connection with the Connectivity Server	205
3.1.3.3	Receiving data	206
3.2	Enumeration types	208
3.2.1	EnumAgentColumnType	208
3.2.2	EnumAgentState (only OSCC V7)	209
3.2.3	EnumCallType	209
3.2.4	EnumDataType	210
3.2.5	EnumErrorCode	210
3.2.6	EnumGroupColumnType	211
3.2.7	EnumLoggedOnMedia	212
3.2.8	EnumMediaType	212
3.2.9	EnumNodeType	212
3.2.10	EnumQueueColumnType	213
3.2.11	EnumServerErrorType	214
3.2.12	EnumServerState	214
3.2.13	EnumUserHandlingState (only OSCC V8)	214
3.2.14	EnumUserPresenceState (only OSCC V8)	215
3.2.15	EnumUserRoutingState (only OSCC V8)	215
3.2.16	EnumValid	215
3.3	Classes	216
3.3.1	IRealtimeLink	216
3.3.1.1	Properties	216
3.3.1.2	Methods	218

3.3.1.3	Events.....	218
3.4	Interfaces.....	222
3.4.1	IAcdContact.....	222
3.4.1.1	Properties.....	222
3.4.2	IAcdContactList.....	222
3.4.2.1	Properties.....	222
3.4.2.2	Methods.....	223
3.4.3	IAcdStep.....	223
3.4.3.1	Properties.....	223
3.4.4	IAcdStepList.....	224
3.4.4.1	Properties.....	224
3.4.4.2	Methods.....	224
3.4.5	IAgentColumnConfig.....	225
3.4.5.1	Properties.....	225
3.4.6	IAgentColumnConfigList.....	227
3.4.6.1	Properties.....	227
3.4.6.2	Methods.....	227
3.4.7	IAgentConfig.....	227
3.4.7.1	Properties.....	227
3.4.8	IAgentObject.....	228
3.4.8.1	Properties.....	228
3.4.9	IAgentObjectList.....	233
3.4.9.1	Properties.....	233
3.4.9.2	Methods.....	233
3.4.10	IAgentsData.....	233
3.4.10.1	Properties.....	233
3.4.11	IColumnConfig.....	234
3.4.11.1	Properties.....	234
3.4.12	IConfiguredViewsEvent.....	236
3.4.12.1	Properties.....	236
3.4.13	IGroupColumnConfig.....	236
3.4.13.1	Properties.....	237
3.4.14	IGroupColumnConfigList.....	238
3.4.14.1	Properties.....	238
3.4.14.2	Methods.....	239
3.4.15	IGroupConfig.....	239
3.4.15.1	Properties.....	239
3.4.16	IGroupObject.....	240
3.4.16.1	Properties.....	240
3.4.17	IGroupObjectList.....	242
3.4.17.1	Properties.....	242
3.4.17.2	Methods.....	243
3.4.18	IGroupData.....	243
3.4.18.1	Properties.....	243
3.4.19	ILogoffResponse.....	244
3.4.19.1	Properties.....	244
3.4.20	ILogonResponse.....	244

Content

3.4.20.1	Properties	244
3.4.21	IQueueColumnConfig	244
3.4.21.1	Properties	245
3.4.22	IQueueColumnConfigList	247
3.4.22.1	Properties	247
3.4.22.2	Methods	247
3.4.23	IQueueConfig	247
3.4.23.1	Properties	247
3.4.24	IQueueObject	248
3.4.24.1	Properties	248
3.4.25	IQueueObjectsList	254
3.4.25.1	Properties	254
3.4.25.2	Methods	254
3.4.26	IQueuesData	254
3.4.26.1	Properties	254
3.4.27	IQueueVirtualAgentObject	255
3.4.27.1	Properties	255
3.4.28	IQueueVirtualAgentObjectsList	256
3.4.28.1	Properties	256
3.4.29	ITeamConfig	257
3.4.29.1	Properties	257
3.4.30	ITeamEntry	257
3.4.30.1	Properties	257
3.4.31	ITeamList	258
3.4.31.1	Properties	258
3.4.31.2	Methods	258
3.4.32	IServerErrorEvent	258
3.4.32.1	Properties	258
3.4.33	IVirtualAgentObject	259
3.4.33.1	Properties	259
3.4.34	IVirtualAgentObjectsList	259
3.4.34.1	Properties	259
3.4.35	IVirtualAgentsData	260
3.4.35.1	Properties	260

1 About this guide

The documentation describes the interface library of the **Integration Connector**.

The CTI interface library is used by application programs with telephony and agent control functions based on the Integration Connector.

The real-time interface library is used to create real-time frontend applications.

This documentation is intended for software developers and project managers.

1.1 Important product information

This guide applies to OSCC-E and the Solution Kit. It contains the entire pallet of the interface library. The following features are not part of the OSCC-E:

- CallCoding
- Journal via Journal Server.
- Voice recording via ASC node.
- Connect balancing (only available for OS Concierge)

1.2 Overview of contents

Section 2.1

Briefly describes the basics needed for application development with the CTI interface library of the OSCC-E server.

At select locations below, examples are given in VisualBasic.

Section 2.2

Explains the enumeration types of the CTI library.

Section 2.3

Describes the "CocCtiConnector" class in detail according to the property procedures, methods and events.

Section 2.4

Describes other classes of the CTI library.

Section 2.5

Describes the interfaces of the CTI library.

Section 3.1

Briefly describes the basics needed for application development with the real-time interface library of the OSCC-E server.

The following subchapters contain application examples in the C# and Java languages.

Section 3.2

Explains the enumeration types of the real-time library.

Section 3.3

Describes the "RealtimeLink" class in detail according to the property procedures, methods and events.

Section 3.4

Describes the interfaces of the real-time library.

1.3 Typographical styles

- *Courier* is used for parameters and constants.
- **Courier bold** is used for methods and event names and for general source code examples and user entries.
- *Courier italics* is used for method and event descriptions for type identifiers.

In the detailed descriptions of methods, parameter types are differentiated as follows:

- Value parameters are presented without special identifying marks (e.g. CallID),
- Return parameters are underscored (e.g. New_CallID),
- Optional parameters are indicated by the subscript "opt" (e.g. Sync_{opt}), and the standard value is presented in the parameter description in square brackets.

2 CTI

2.1 Basics

Important note in advance:

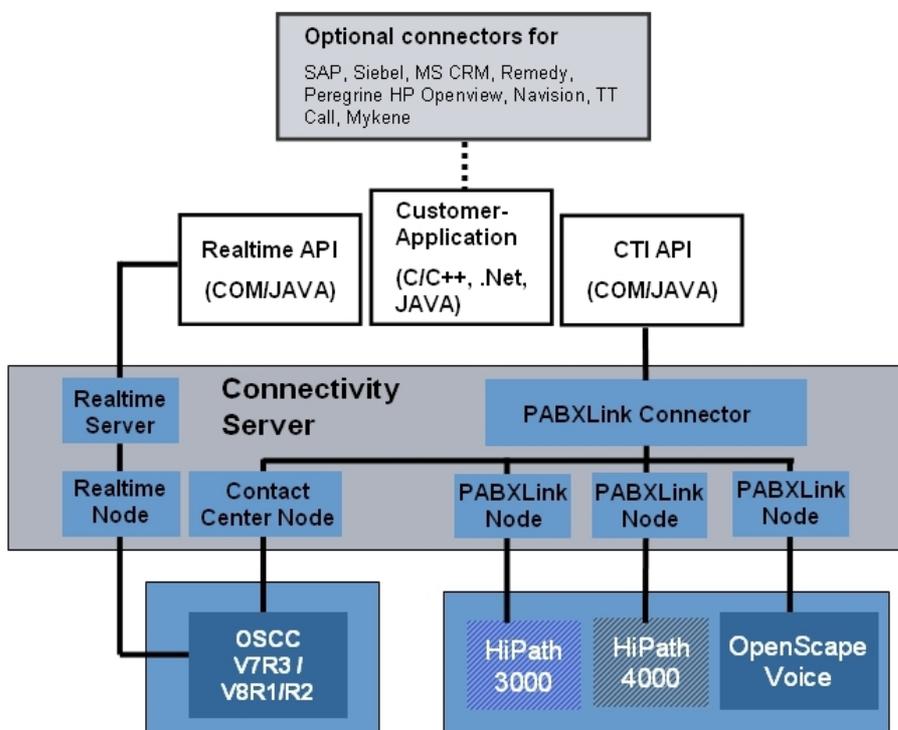
As the DLL can only be instantiated once, server applications are not programmable with it.

2.1.1 System architecture

The OSCC-E server manages one or more telephone systems and makes functions available for the control of **extensions** and **agents**. These functions can be used in application programs via the **interface library**.

The interface library is a COM DLL that makes the CTI and agent control functions of the OSCC-E server available. The interface library can therefore be used in any programming language that exists for a COM implementation, e.g. Visual Basic®, Visual C#® and Delphi®.

As shown in the following figure by way of example, the OSCC-E server and the interface library are components of a three-layer CTI system architecture. This architecture can incorporate different **telephone systems** and **call centers/contact centers** at the switch level and contains the components of the OSCC-E server at the interface level, which is then addressed on the application level via the interface library.



As the figure shows, the OSCC-E server communicates with each telephone system via a so-called **node**. A node is uniquely assigned to one telephone system and is identified by a node name. This node communicates with this system via the CSTA protocol.

If the telephone system is additionally connected with a call center/contact center – which is usually the case – the control functions are likewise implemented via the node, namely via the call center SDK functions. Therefore, any necessary version adaptations in the contact center and the associated SDK no longer require adaptations in the client application.

In the client application, each connection object – an object of the "CustPABXCtiLink" class described below – of the interface library is assigned to exactly one node.

The interface library communicates with the OSCC-E server via messages using TCP/IP. For this reason, the client system must support socket connection to the OSCC-E server via a defined port.

2.1.2 Application development

IMPORTANT NOTE:

The interface library offers a range of CTI functions comparable to that of the *OpenScape Contact Center Desktop* which can be used in development projects.

The CTI functions and events of the interface library are based on the functions and messages of the CSTA interface of the telephone system. Signaling and the scope of features of this interface differ from the scope of features of HFA end devices (*HiPath Feature Access*), among other things in passive call situations and conferencing and transfer situations. Therefore, depending on the particular situation, it is possible that the functions and information offered via the interface library may differ from those offered on an end device.

2.1.2.1 Registration of the interface library

Before the interface library can be used, it must be registered in the operating system. This can take place in the development environment or directly under Windows with the following command, for example:

```
regsvr32.exe coccustcticonlib.dll
```

2.1.2.2 Application examples

The examples provided below are written in C#. Usage is similar in other COM-compatible programming languages.

Declaring a connection object

```
using coccustcticonlib;  
  
private CocCtiConnector ctiConnector;
```

Instantiation of a connection object

The connection object must be instantiated before use.

```
ctiConnector = new CocCtiConnector();  
establishment  
of a connection to the OSCC-E server
```

The first step after instantiation is to establish the connection to the OSCC-E server. At least the following connection parameters must be specified (the values given here are examples only).

```
this.ctiConnector.ConfigData.CtiConnectivityConfigData.IpAddress = „192.168.0.0“;
this.ctiConnector.ConfigData.CtiConnectivityConfigData.IpAddress = „17010“;
this.ctiConnector.ConfigData.CtiConnectivityConfigData.SiteId = „LOC001“;
this.ctiConnector.ConfigData.CtiConnectivityConfigData.Extension = „1234“;
this.ctiConnector.ConfigData.CtiConnectivityConfigData.ConnectAndMonitor = true;
```

```
ErrorEnum result = this.ctiConnector.Open(false);
```

```
if (result == ErrorEnum.glbOK)
{
    this.ctiConnector.ConfigData.CtiConnectivityConfigData.ConnectAndMonitor = true;
}
```

Within an application only one connection object can be created. If connection was successful, the **Open** function returns the value `glbOK`. If not, an error occurred, and an error description can be retrieved with the **GetErrorAsString** function (see below).

Terminating the connection

When the program ends, the connection to the OSCC-E server should be terminated and the reference to the object should be released.

```
this.ctiConnector.Close();
this.ctiConnector = null;
```

2.1.2.3 Synchronous and asynchronous method execution

Control functions for extensions and agents send internal messages to the OSCC-E server and evaluate the responses. Therefore, these control functions can be executed either synchronously or asynchronously.

- In the case of **synchronous execution**, the method is not ended until the corresponding replies from the OSCC-E server have arrived and all results are available. When the method is ended, values are assigned to the return parameters and the method result (error or no error) is determined on the basis of the server replies.
- In the case of the **asynchronous execution**, the method is ended after the messages are sent to the OSCC-E server without waiting for its reply. When the method is ended, no values are assigned to the return parameters, which is why literal values can be assigned to these parameters when the method is called up. The method result has the value `glbOK` (no error if the parameter test did not identify errors directly after the method was called up). The event belonging to the method

execution then delivers the method result and the values of the return parameters (e.g. the **OnMakeCallEvent** event for the **MakeCall** method). The method execution mode is controlled via a parameter.

Synchronous execution is useful for simple applications in which the method result is associated with immediate replies or changes at the application's user interface (e.g. a change in the state of a button). It is also useful when multiple method calls need to occur in sequence in relation to the individual events.

NOTE:

In complex applications with a higher event rate or connection parallelism, asynchronous method execution is preferable.

The optional parameters of the functions can be used in VB.NET but not in C#. The standard value for the optional parameter **Sync** is false, i.e. the method is executed asynchronously by default.

Commands have to be called and events have to be handled within the application thread created by the connection object.

2.1.2.4 Troubleshooting

Errors relating to method execution are output in two different ways:

- Through the result of a method when called synchronously (in case of an error, the result is not equal to glbOK),

OR

- Through the result data of an event when called asynchronously.

All other errors are indicated by

- OnErrorEvent.

For each error code, a written description for the display or the log file output can be retrieved using the **GetErrorAsString** method.

If the error code indicates a **CSTA error**, the **CstaError** property delivers a description of the error according to its error class and error value.

In addition to the error codes that can be returned by various methods, there are specific error codes that are individually listed below in the method descriptions.

2.1.2.5 Status changes among extensions and agents

The status of extensions and agents can change due to actions of the application program or through external events (for example, through an incoming call).

Changes in the extension status are indicated by the **OnPhoneStateChangeInfoEvent** event, while changes in the agent status are indicated by the **OnAcdUserStateChangeInfoEvent** event.

After the successful call of a CTI command the application should not allow any other CTI command until the following **OnPhoneStateChangeEvent** notifies the now permitted commands.

2.2 Enumeration types

2.2.1 AcdContactTypeEnum

This enumeration type describes the media type and the assignment method of a contact in the call center.

2.2.2 AcdDequeueReasonEnum

This enumeration type is intended for future functions.

2.2.3 AcdFunctionTypeEnum

This enumeration type describes the special ACD functions.

2.2.4 AcdHandlingStateEnum

This enumeration type describes the handling state of a contact in the call center.

2.2.5 AcdMediaTypeEnum

This enumeration type describes the media type of a contact.

2.2.6 AcdPartyTypeEnum

This enumeration type describes the type of subscriber to which an ACD call is currently assigned.

2.2.7 AgentStateOsvHuntgroupEnum

This enumeration type describes the availability status of an agent in a hunt group.

2.2.8 ApplicationIdentifierEnum

This enumeration type describes the type of application in relation to use of the interface library.

2.2.9 AutoAnswerFlagEnum

This enumeration type describes the capability of an analog or digital end device to dial with the handset hung up.

2.2.10 BusyStateDetailEnum

This enumeration type describes details on the busy status of an extension.

2.2.11 BusyStateEnum

This enumeration type describes the busy status of an extension.

2.2.12 CallDirectionEnum

This enumeration type describes the direction states of a call.

2.2.13 CallListChangeNotificationEnum

This enumeration type describes a change in a call connected with an extension.

2.2.14 CallStateEnum

This enumeration type describes the states of a call.

2.2.15 ConferenceEventTypeEnum

This enumeration type describes a change in the state of a conference.

2.2.16 CstaVersionEnum

This enumeration type describes the version of the underlying CSTA.

2.2.17 DeviceTypeEnum

This enumeration type describes a device type.

2.2.18 DoNotDisturbFlagEnum

This enumeration type describes the do-not-disturb state of an extension.

2.2.19 ErrorEnum

This enumeration type describes error codes (glbOK means "no error").

2.2.20 EventCauseEnum

This enumeration type describes the event reason for a CSTA command.

2.2.21 EventTypeEnum

This enumeration type is intended for future functions.

2.2.22 ForwardingStateEnum

This enumeration type is intended for future functions.

2.2.23 IPVersionEnum

This enumeration type describes the IP protocol version used.

2.2.24 LocalConnectionStateEnum

This enumeration type describes the current state of a connection.

2.2.25 LogLevelEnum

This enumeration type describes the level of a log message.

2.2.26 LogMessageDelimiterEnum

This enumeration type indicates the delimiter between individual values in log messages.

2.2.27 MonitorDeviceHandlingFlagEnum

This enumeration type describes the type of monitor handling.

2.2.28 MonitorStateEnum

This enumeration type describes the monitor state.

2.2.29 NodeTypeEnum

This enumeration type describes the node type (telephone system, ACD, etc.).

2.2.30 ServerConnectionModeEnum

This enumeration type describes the connection mode.

2.2.31 ServerConnectionStateEnum

This enumeration type describes the connection state to the OSCC-E server.

2.2.32 ServerTypeEnum

This enumeration type describes the type of OSCC-E server.

2.2.33 SetForwardingActionEnum

This enumeration type describes the state of the call forwarding.

2.2.34 SiteStateEnum

This enumeration type describes the state of a node.

2.2.35 SiteTypeEnum

This enumeration type describes the node type.

2.2.36 UserAgentStateEnum

This enumeration type describes the routing state in which an agent may be found.

2.3 "CocCtiConnector" class

An object of the **CocCtiConnector** class makes a connection to the OSCCE server available via an extension and for an agent registered there.

2.3.1 Properties

2.3.1.1 AvailableLicences

This property contains the number of free licenses for the application type at the time when the connection was established.

AvailableLicences	
<i>Integer</i>	

2.3.1.2 ConfigData

This property contains all configuration and setting data.

ConfigData	
<i>IConfigDataX</i>	

2.3.1.3 ServerConnectionMode

This property contains the server connection mode.

ServerConnectionMode	
<i>ServerConnectionModeEnum</i>	

2.3.1.4 ServerConnectionState

This property contains the server connection state.

ServerConnectionState	
<i>ServerConnectionStateEnum</i>	

2.3.2 Methods

2.3.2.1 AcdAvailable

This method sets the agent for the media type to the "available" state.

Table: AcdAvailable

AcdAvailable		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
UserData	<i>String</i>	Additional user data

AcdAvailable		
MediaType	<i>AcdMediaTypeEnum</i>	Media type
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdAvailableDataX</i>		
<i>Events</i>		
<i>OnAcdAvailableEvent, OnAcdUserStateChangeInfoEvent</i>		
<i>Comment</i>		
MediaType must not be AcdMediaTypeEnum.amtAll.		

2.3.2.2 AcdCheckPassword

This method checks the validity of the password for a user ID.

Table: AcdCheckPassword

AcdCheckPassword		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
Password	<i>String</i>	Password
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdCheckPasswordDataX</i>		
<i>Events</i>		
<i>OnAcdCheckPasswordEvent</i>		
<i>Comment</i>		
This function can only be called up after the AcdRegisterForGlobalEvents function was executed successfully.		

2.3.2.3 AcdExitPostProcessingState

This method ends post-processing.

Table: AcdExitPostProcessingState

AcdAvailable		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
AcdCallId	<i>String</i>	Contact ID
RequeueCount	<i>Integer</i>	Number of requeues
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		

CTI
 "CocCtiConnector" class

AcdAvailable		
<i>IAcdExitPostProcessingStateDataX</i>		
<i>Events</i>		
<i>OnAcdExitPostProcessingStateEvent, OnAcdUserStateChangeInfoEvent</i>		

2.3.2.4 AcdGetBusinessUnitForExtension

This method delivers the business unit for which an extension is set up.

Table: AcdGetBusinessUnitForExtension

AcdGetBusinessUnitForExtension		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
Extension	<i>String</i>	Extension
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdGetBusinessUnitForExtensionDataX</i>		
<i>Events</i>		
<i>OnAcdGetBusinessUnitForExtensionEvent</i>		

2.3.2.5 AcdGetBusinessUnitForUserId

This method delivers the business unit to which an agent belongs.

Table: AcdGetBusinessUnitForUserId

AcdGetBusinessUnitForUserId		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdGetBusinessUnitForUserIdDataX</i>		
<i>Events</i>		
<i>OnAcdGetBusinessUnitForUserIdEvent</i>		

2.3.2.6 AcdGetBusinessUnits

This method delivers the setup business units.

Table: AcdGetBusinessUnits

AcdGetBusinessUnits		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserData	<i>String</i>	Additional user data

AcdGetBusinessUnits		
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdGetBusinessUnitsDataX</i>	<i>Result data</i>	
<i>Events</i>		
OnAcdGetBusinessUnitsEvent		
<i>Comment</i>		
This message is also available in the restricted connection mode.		

2.3.2.7 AcdGetCallInformation

This method delivers information on an ACD voice call.

Table: AcdGetCallInformation

AcdGetCallInformation		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
AcdCallId	<i>String</i>	ACD call ID
PabxCallId	<i>String</i>	Call ID
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdGetCallInformationDataX</i>		
<i>Events</i>		
OnAcdGetCallInformationEvent		
<i>Comment</i>		
AcdCallId or PabxCallId must contain a value.		

2.3.2.8 AcdGetPostProcessingReasons

This method delivers the post-processing activity reasons of the business unit.

Table: AcdGetPostProcessingReasons

AcdGetPostProcessingReasons		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
BusinessUnitName	<i>String</i>	Business unit
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IGetPostProcessingReasonsDataX</i>		
<i>Events</i>		

CTI
 "CocCtiConnector" class

AcidGetPostProcessingReasons
OnAcidGetPostProcessingReasons
<i>Comment</i>
This method requires OSCC V8 R0 or higher.

2.3.2.9 AcidGetPostProcessingState

This method delivers the processing states of all active contacts of a user.

Table: AcidGetPostProcessingState

AcidGetPostProcessingState		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserID	<i>String</i>	User ID
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcidGetPostProcessingStateDataX</i>		
<i>Events</i>		
OnAcidGetPostProcessingReasonsEvent		
<i>Comment</i>		
This method requires OSCC V8 R0 or higher.		

2.3.2.10 AcidGetQueues

This method delivers the queues set up in a business unit for a media type.

Table: AcidGetQueues

AcidGetQueues		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
MediaType	<i>AcidMediaTypeEnum</i>	Media type
BusinessUnit	<i>String</i>	Business unit
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcidGetQueuesDataX</i>	<i>Result data</i>	
<i>Events</i>		
OnAcidGetQueuesEvent		

2.3.2.11 AcdGetTeams

This method delivers the teams set up in a business unit.

Table: AcdGetTeams

AcdGetTeams		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
BusinessUnit	<i>String</i>	Business unit
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdGetTeamsDataX</i>	<i>Result data</i>	
<i>Events</i>		
OnAcdGetTeamsEvent		

2.3.2.12 AcdGetUnavailableReasons

This method delivers the unavailability reasons defined in a business unit.

Table: AcdGetUnavailableReasons

AcdGetUnavailableReasons		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
BusinessUnitName	<i>String</i>	Business unit
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdGetUnavailableReasonsDataX</i>	<i>Result data</i>	
<i>Events</i>		
OnAcdGetUnavailableReasonsEvent		

2.3.2.13 AcdGetUserIdForUserName

This method delivers the user ID for an ACD user name.

Table: AcdGetUserIdForUserName

AcdGetUserIdForUserName		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserName	<i>String</i>	User name
BusinessUnitName	<i>String</i>	Business unit
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		

CTI
 "CocCtiConnector" class

AcdGetUserIdForUserName	
<i>IGetUserIdForUserNameDataX</i>	
<i>Events</i>	
OnAcdGetUserIdForUserName	

2.3.2.14 AcdGetUsers

This method delivers the users assigned to a business unit for a media type.

Table: AcdGetUsers

AcdGetUsers		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
MediaType	<i>AcdMediaTypeEnum</i>	Media type
BusinessUnit	<i>String</i>	Business unit
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdGetUsersDataX</i>	<i>Result data</i>	
<i>Events</i>		
OnAcdGetUsersEvent		
<i>Comment</i>		
This message is also available in the restricted connection mode.		

2.3.2.15 AcdGetUserState

This method delivers the ACD state of a user.

Table: AcdGetUserState

AcdGetUserState		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
UserData	<i>String</i>	Additional user data
MediaType	<i>AcdMediaTypeEnum</i>	Media type
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdGetUserStateDataX</i>	<i>Result data</i>	
<i>Events</i>		
OnAcdGetUserStateEvent		

2.3.2.16 AcdGetWorkReasons

This method delivers the work reasons defined in a business unit.

Table: AcdGetWorkReasons

AcdGetWorkReasons		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
BusinessUnitName	<i>String</i>	Business unit
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdGetWorkReasonsDataX</i>	<i>Result data</i>	
<i>Events</i>		
OnAcdGetWorkReasonsEvent		

2.3.2.17 AcdGetWrapupReasons

This method delivers the wrapup reasons defined in a business unit.

Table: AcdGetWrapupReasons

AcdGetWrapupReasons		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
BusinessUnitName	<i>String</i>	Business unit
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdGetWrapupReasonsDataX</i>	<i>Result data</i>	
<i>Events</i>		
OnAcdGetWrapupReasonsEvent		

2.3.2.18 AcdLogoff

This method logs off the agent for a media type.

Table: AcdLogoff

AcdLogoff		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
UserData	<i>String</i>	Additional user data
MediaType	<i>AcdMediaTypeEnum</i>	Media type
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		

CTI
 "CocCtiConnector" class

AcdLogoff	
<i>IAcdLogoffDataX</i>	
<i>Events</i>	
OnAcdLogoffEvent, OnAcdUserStateChangeInfoEvent	
<i>Comment</i>	
MediaType must not be AcdMediaTypeEnum.amtAll.	

2.3.2.19 AcdLogon

This method logs on the agent for a media type.

Table: AcdLogon

AcdLogon		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
UserData	<i>String</i>	Additional user data
MediaType	<i>AcdMediaTypeEnum</i>	Media type
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdLogonDataX</i>		
<i>Events</i>		
OnAcdLogonEvent, OnAcdUserStateChangeInfoEvent		
<i>Comment</i>		
MediaType must not be AcdMediaTypeEnum.amtAll.		

2.3.2.20 AcdRegisterForFunctions

This method registers an agent for special ACD functions.

Table: AcdRegisterForFunctions

AcdRegisterForFunctions		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
UserData	<i>String</i>	Additional user data
FunctionTypeList	<i>AcdContactDataListParamX</i>	List of special functions
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdRegisterForFunctionsDataX</i>		
<i>Events</i>		
OnAcdRegisterForFunctionsEvent		

2.3.2.21 AcdRegisterForGlobalEvents

This method registers for general ACD events.

Table: AcdRegisterForGlobalEvents

AcdRegisterForGlobalEvents		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdRegisterForGlobalEventsDataX</i>		
<i>Events</i>		
<i>OnAcdRegisterForGlobalEventsEvent</i>		

2.3.2.22 AcdRegisterForNodeEvents

This method registers for the media-related events of an agent.

Table: AcdRegisterForNodeEvents

AcdRegisterForNodeEvents		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
UserData	<i>String</i>	Additional user data
MediaType	<i>AcdMediaTypeEnum</i>	Media type
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdRegisterForNodeEventsDataX</i>		
<i>Events</i>		
<i>OnAcdRegisterForNodeEventsEvent, OnAcdUserStateChangeInfoEvent, OnAcdUserStateInfoEvent</i>		
<i>Comment</i>		
MediaType must not be AcdMediaTypeEnum.amtAll.		

2.3.2.23 AcdSaveWrapupData

This method saves the wrapup reasons for a contact.

Table: AcdSaveWrapupData

AcdSaveWrapupData		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
AcdCallId	<i>String</i>	ACD contact ID

CTI
 "CocCtiConnector" class

AcdSaveWrapupData		
WrapupReasonKeyList	<i>AcdWrapupReasonKeyDataListParamX</i>	List of keys of the wrapup reasons
QueueKey	<i>Integer</i>	Queue key
RequeueCount	<i>Integer</i>	Number of requeues
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdSaveWrapupDataX</i>		
<i>Events</i>		
OnAcdSaveWrapupDataEvent		

2.3.2.24 AcdSetPostProcessingState

This method makes it possible to change the reason for the post-processing of a contact.

Table: AcdSetPostProcessingState

AcdSetPostProcessingState		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
ReasonKey	<i>Integer</i>	Reason key
AcdCallId	<i>String</i>	ACD contact ID
RequeueCount	<i>Integer</i>	Number of requeues
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdSetPostProcessingStateDataX</i>		
<i>Events</i>		
OnAcdSetPostProcssingStateEvent		
<i>Comment</i>		
This method requires OSCC V8 R0 or higher.		

2.3.2.25 AcdSetPrimaryContact

This method defines the contact as the primary contact, i.e. the contact currently being processed.

Table: AcdSetPrimaryContact

AcdSetPrimaryContact		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
AcdCallId	<i>String</i>	ACD contact ID
RequeueCount	<i>Integer</i>	Number of requeues
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdSetPrimaryContactDataX</i>		
<i>Events</i>		
<i>OnAcdSetPrimaryContactEvent</i>		
<i>Comment</i>		
This method requires OSCC V8 R1 or higher.		

2.3.2.26 AcdUnavailable

This method sets the agent for a media type to the "unavailable" state.

Table: AcdUnavailable

AcdUnavailable		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
UserData	<i>String</i>	Additional user data
MediaType	<i>AcdMediaTypeEnum</i>	Media type
ReasonCode	<i>Integer</i>	Unavailability reasons
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdUnavailableDataX</i>		
<i>Events</i>		
<i>OnAcdUnavailableEvent, OnAcdUserStateChangeInfoEvent</i>		
<i>Comment</i>		
MediaType must not be AcdMediaTypeEnum.amtAll.		

2.3.2.27 AcdUnregisterForFunctions

This method deregisters an agent for special ACD functions.

Table: AcdUnregisterForFunctions

AcdRegisterForFunctions		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
UserData	<i>String</i>	Additional user data
FunctionTypeList	<i>AcdFunctionTypeListParam</i>	List of special functions
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdUnregisterForFunctionsDataX</i>		
<i>Events</i>		
<i>OnAcdUnregisterForFunctionsEvent</i>		

2.3.2.28 AcdUnregisterForGlobalEvents

This method deregisters for general ACD events.

Table: AcdUnregisterForGlobalEvents

AcdUnregisterForGlobalEvents		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdUnregisterForGlobalEventsDataX</i>		
<i>Events</i>		
<i>OnAcdUnregisterForGlobalEventsEvent</i>		

2.3.2.29 AcdUnregisterForNodeEvents

This method deregisters for the media-related events of an agent.

Table: AcdUnregisterForNodeEvents

AcdUnregisterForNodeEvents		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
UserData	<i>String</i>	Additional user data
MediaType	<i>AcdMediaTypeEnum</i>	Media type
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		

AcdUnregisterForNodeEvents	
<i>IAcdUnregisterForNodeEventsDataX</i>	
<i>Events</i>	
OnAcdUnregisterForNodeEventsEvent	
<i>Comment</i>	
MediaType must not be AcdMediaTypeEnum.amtAll.	

2.3.2.30 AcdWork

This method sets the agent for a media type to the "work" state.

Table: AcdWork

AcdWork		
<i>Parameter</i>		
LocationId	<i>String</i>	Node name
UserId	<i>String</i>	User ID
UserData	<i>String</i>	Additional user data
MediaType	<i>AcdMediaTypeEnum</i>	Media type
ReasonCode	<i>Integer</i>	Work reason
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAcdWorkDataX</i>		
<i>Events</i>		
OnAcdWorkEvent, OnAcdUserStateChangeInfoEvent		
<i>Comment</i>		
MediaType must not be AcdMediaTypeEnum.amtAll.		

2.3.2.31 AgentAvailable

This method sets the agent state of an extension to the "available" state.

Table: AgentAvailable

AgentAvailable		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAgentAvailableDataX</i>		
<i>Events</i>		
OnAgentAvailableEvent, OnAgentStateChangeInfoEvent		
<i>Comment</i>		

AgentAvailable
This method is used to modify the agent state of an extension administrated in the HiPath 4000. This agent state permits an ACD routing independent of the OpenScape ContactCenter and is primarily used for emergency routing. If the agent ID corresponds to the registered OSCC user ID, this also changes the OSCC routing state and this change is displayed in the OnAcidUserStateChangeInfoEvent.

2.3.2.32 AgentLogoff

This method sets the agent state of an extension to the "logged off" state.

Table: AgentLogoff

AgentLogoff		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAgentLogoffDataX</i>		
<i>Events</i>		
OnAgentLogoffEvent, OnAgentStateChangeInfoEvent		
<i>Comment</i>		
This method is used to modify the agent state of an extension administrated in the HiPath 4000. This agent state permits an ACD routing independent of the OpenScape ContactCenter and is primarily used for emergency routing. If the agent ID corresponds to the registered OSCC user ID, this also changes the OSCC routing state and this change is displayed in the OnAcidUserStateChangeInfoEvent.		

2.3.2.33 AgentLogon

This method sets the agent state of an extension to the "logged on" state.

Table: AgentLogon

AgentLogon		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAgentLogonDataX</i>		
<i>Events</i>		
OnAgentLogonEvent, OnAgentStateChangeInfoEvent		
<i>Comment</i>		
This method is used to modify the agent state of an extension administrated in the HiPath 4000. This agent state permits an ACD routing independent of the OpenScape ContactCenter and is primarily used for emergency routing. If the agent ID corresponds to the registered OSCC user ID, this also changes the OSCC routing state and this change is displayed in the OnAcidUserStateChangeInfoEvent.		

2.3.2.34 AgentUnavailable

This method sets the agent state of an extension to the "available" state.

Table: AgentUnavailable

AgentUnavailable		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAgentUnavailableDataX</i>		
<i>Events</i>		
<i>OnAgentUnavailableEvent, OnAgentStateChangeInfoEvent</i>		
<i>Comment</i>		
This method is used to modify the agent state of an extension administrated in the HiPath 4000. This agent state permits an ACD routing independent of the OpenScape ContactCenter and is primarily used for emergency routing. If the agent ID corresponds to the registered OSCC user ID, this also changes the OSCC routing state and this change is displayed in the OnAcidUserStateChangeInfoEvent.		

2.3.2.35 AgentWork

This method sets the agent state of an extension to the "work" state.

Table: AgentWork

AgentWork		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAgentWorkDataX</i>		
<i>Events</i>		
<i>OnAgentWorkEvent, OnAgentStateChangeInfoEvent</i>		
<i>Comment</i>		
This method is used to modify the agent state of an extension administrated in the HiPath 4000. This agent state permits an ACD routing independent of the OpenScape ContactCenter and is primarily used for emergency routing. If the agent ID corresponds to the registered OSCC user ID, this also changes the OSCC routing state and this change is displayed in the OnAcidUserStateChangeInfoEvent.		

2.3.2.36 AlternateCall

This method alternates between a current call and a call on hold.

Table: AlternateCall

AlternateCall		
<i>Parameter</i>		
Extension	<i>String</i>	Extension

CTI
 "CocCtiConnector" class

AlternateCall		
CallId	<i>String</i>	ID of call
CallIdOnHold	<i>String</i>	ID of call on hold
Sync opt	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAAlternateCallDataX</i>	<i>Result data</i>	
<i>Events</i>		
OnAlternateCallEvent		
<i>Comment</i>		
If an empty string is passed to each of the Extension, CallId and CallIdOnHold parameters, toggling occurs between the current call and the call on hold.		

2.3.2.37 AnswerCall

This method answers a call.

Table: AnswerCall

AnswerCall		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallId	<i>String</i>	ID of call
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IAnswerCallDataX</i>	<i>Result data</i>	
<i>Events</i>		
OnAnswerCallEvent, OnPhoneStateChangeInfoEvent		
<i>Comment</i>		
If an empty string is passed to each of the Extension and CallId parameters, the current call is accepted.		

2.3.2.38 CheckMonitoringDevice

This method checks the monitoring status of an extension.

Table: CheckMonitoringDevice

CheckMonitoringDevice		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>ICheckMonitoringDeviceDataX</i>	<i>Result data</i>	
<i>Events</i>		
OnCheckMonitoringDeviceEvent		

2.3.2.39 ClearConnection

This method terminates a call.

Table: ClearConnection

ClearConnection		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallId	<i>String</i>	ID of call
Trunc _{opt}	<i>String</i>	Trunk number of an external connection [""]
Sync _{opt}	<i>Boolean</i>	Synchronous execution [true]
<i>Return</i>		
<i>IClearConnectionDataX</i>	<i>Result data</i>	
<i>Events</i>		
<i>OnClearConnectionEvent, OnPhoneStateChangeInfoEvent</i>		
<i>Comment</i>		
If an empty string is passed to each of the Extension, CallId and Trunc parameters, the current call is terminated.		

2.3.2.40 Close

This method closes the current connection to the OSCC-E server.

2.3.2.41 ConferenceCall

This method connects a call on hold with another call to form a conference call.

Table: ConferenceCall

ConferenceCall		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallId	<i>String</i>	ID of call
CallIdOnHold	<i>String</i>	ID of call on hold
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IConferenceCallDataX</i>	<i>Result data</i>	
<i>Events</i>		
<i>OnConferenceCallEvent, OnConferenceInfoEvent, OnPhoneStateChangeInfoEvent</i>		
<i>Comment</i>		
If an empty string is passed to each of the Extension, CallId and CallIdOnHold parameters, the call on hold is added to the conference.		

CTI
 "CocCtiConnector" class

2.3.2.42 ConsultCall

This method places a call on hold and initiates a consultation call.

Table: ConsultCall

ConsultCall		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallId	<i>String</i>	ID of call
Destination	<i>String</i>	Destination of consultation call
AutoAnswerFlag	<i>AutoAnswerFlagEnum</i>	Dial while handset is hung up
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IConsultCallDataX</i>	<i>Result data</i>	
<i>Events</i>		
<i>OnConsultCallEvent, OnPhoneStateChangeInfoEvent</i>		
<i>Comment</i>		
If an empty string is passed to each of the Extension and CallId parameters, a consultation call is established for the current call.		

2.3.2.43 CreateCallback

This method sets up a callback request at the extension for a call that could not be accepted by the called extension due to its state.

Table: CreateCallback

CreateCallback		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallId	<i>String</i>	ID of call
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>ICreateCallbackDataX</i>	<i>Result data</i>	
<i>Events</i>		
<i>OnCreateCallbackEvent, OnPhoneStateChangeInfoEvent</i>		
<i>Comment</i>		
The callback takes place from the telephone system to the extension as soon as the called extension is ready to receive a call.		

2.3.2.44 DeflectCall

This method deflects a call to another destination number.

Table: DeflectCall

DeflectCall		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallID	<i>String</i>	ID of call
Destination	<i>String</i>	Deflection destination
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IDeflectCallDataX</i>	<i>Result data</i>	
<i>Events</i>		
<i>OnDeflectCallEvent, OnPhoneStateChangeInfoEvent</i>		
<i>Comment</i>		
<p>If an empty string is passed to each of the Extension and CallId parameters, the current call is deflected. If an extension is in the idle state and a ringing call should be accepted by another extension, the other extension should be entered under "Extension" and the own extension should be entered under "Destination".</p>		

2.3.2.45 DeleteCallback

This method deletes a callback request to another extension.

Table: DeleteCallback

DeleteCallback		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
Destination	<i>String</i>	Destination extension
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IDeleteCallbackDataX</i>	<i>Result data</i>	
<i>Events</i>		
<i>OnDeleteCallbackEvent</i>		

2.3.2.46 DialDigits

This method dials a number sequence in cut-through dialing (non-block dialing).

Table: DialDigits

DialDigits		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallId	<i>String</i>	ID of call

CTI
 "CocCtiConnector" class

DialDigits		
Digits	<i>String</i>	Number sequence
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>IDialDigitsDataX</i>		
<i>Long</i>	<i>0 = no error</i>	
<i>Events</i>		
OnDialDigitsEvent		

2.3.2.47 GetAgentState

This method checks the agent status of an extension.

Table: GetAgentState

GetAgentState		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IGetAgentStateDataX</i>		
<i>Events</i>		
OnFailoverInfoEvent		

2.3.2.48 GetAgentStateOsvHuntgroup

This method checks the agent status of a hunt group extension.

Table: GetAgentStateOsvHuntgroup

GetAgentStateOsvHuntgroup		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IGetAgentStateOsvHuntgroupDataX</i>		
<i>Events</i>		
OnGetAgentStateOsvHuntgroupEvent		

2.3.2.49 GetDoNotDisturb

This method checks the do-not-disturb state of the logged-on extension.

Table: GetDoNotDisturb

GetDoNotDisturb		
<i>Parameter</i>		
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IGetDoNotDisturbDataX</i>		

GetDoNotDisturb
<i>Events</i>
OnGetDoNotDisturbEvent

2.3.2.50 GetErrorAsString

This method delivers a descriptive text for an error code.

Table: GetErrorAsString

GetErrorAsString		
<i>Parameter</i>		
Errorcode	<i>ErrorEnum</i>	Error code
<i>Return</i>		
<i>String</i>		

2.3.2.51 GetExtensionCallList

This method delivers the calls currently connected with an extension.

Table: GetExtensionCallList

GetExtensionCallList		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
<i>Return</i>		
<i>IGetExtensionCallListDataX</i>		

2.3.2.52 GetForwarding

This method checks the forwarding state of the logged-on extension.

Table: GetForwarding

GetForwarding		
<i>Parameter</i>		
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IGetForwardingDataX</i>		
<i>Events</i>		
OnGetForwardingEvent		

2.3.2.53 GetPhoneState

This method delivers the status of an extension.

Table: GetPhoneState

GetForwarding		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
<i>Return</i>		

CTI
 "CocCtiConnector" class

GetForwarding	
<i>IGetPhoneStateDataX</i>	

2.3.2.54 HoldCall

This method places a connection on hold (during which the caller usually hears a message on hold or music on hold through the telephone system).

Table: HoldCall

HoldCall		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallId	<i>String</i>	ID of call
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IHoldCallDataX</i>		
<i>Events</i>		
<i>OnHoldCallEvent, OnPhoneStateChangeInfoEvent</i>		
<i>Comment</i>		
If an empty string is passed to the Extension parameter, the current call is placed on hold.		

2.3.2.55 IsExtensionOutOfService

This checks whether an extension is out of service.

Table: IsExtensionOutOfService

IsExtensionOutOfService		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
<i>Return</i>		
<i>Boolean</i>		

2.3.2.56 MakeCall

This method initiates an outgoing call.

Table: MakeCall

MakeCall		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
Destination	<i>String</i>	Destination number
AutoAnswerFlag	EnumAutoAnswerFlag	Dial while handset is hung up
NumberToDisplay _{opt}	<i>String</i>	Transmitted phone number [""]
NameToDisplay _{opt}	<i>String</i>	Transmitted name [""]
Sync _{opt}	<i>Boolean</i>	Synchronous execution optional [true]
<i>Return</i>		
<i>IMakeCallDataX</i>		

MakeCall
<i>Events</i>
<i>OnMakeCallEvent, OnPhoneStateChangeInfoEvent, OnPhoneDialingInfoEvent</i>
<i>Comment</i>
If an empty string is transferred to the Extension parameter, the call is initiated from the current extension. The usability of the NumberToDisplay and NameToDisplay parameters depends on the settings of the telephone system.

2.3.2.57 Open

This method opens a connection to the OSCC-E server.

Table: Open

Open		
<i>Parameter</i>		
Restricted	<i>Boolean</i>	Open a connection with restricted functionality
<i>Return</i>		
<i>ErrorEnum</i>		
<i>Comment</i>		
Before this method is called, the IpAddress , Port , Siteld , Extension and ApplicationIdentifier properties must be assigned valid values in the ConfigData. If Restricted is true, only those functions and properties are available that are required prior to a fully valid connection, for example during a user logon. An Extension does not need to be set for this logon.		

2.3.2.58 ReconnectCall

This method terminates a consultation call and returns to the call on hold.

Table: ReconnectCall

ReconnectCall		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallId	<i>String</i>	ID of call
CallIdOnHold	<i>String</i>	ID of call on hold
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IReconnectCallDataX</i>		
<i>Events</i>		
<i>OnReconnectCallEvent, OnPhoneStateChangeInfoEvent</i>		
<i>Comment</i>		
If an empty string is passed to each of the Extension, CallId and CallIdOnHold parameters, you return to the current call.		

CTI
 "CocCtiConnector" class

2.3.2.59 RetrieveCall

This method retrieves a call on hold.

Table: RetrieveCall

RetrieveCall		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallId	<i>String</i>	ID of call
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IRetrieveCallDataX</i>		
<i>Events</i>		
<i>OnRetrieveCallEvent, OnPhoneStateChangeInfoEvent</i>		
<i>Comment</i>		
If an empty string is passed to each of the Extension and CallId parameters, you return to the current call.		

2.3.2.60 SendDTMF

This method sends a command sequence using DTMF dialing within an active call.

Table: SendDTMF

SendDTMF		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallId	<i>String</i>	ID of call
Digits	<i>String</i>	Command sequence
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>ISendDTMFDataX</i>		
<i>Events</i>		
<i>OnSendDTMFEvent</i>		
<i>Comment</i>		
The command sequence may only contain the digits "0" to "9", the letters "A" to "D" and the symbols "*" and "#".		

2.3.2.61 SetAgentStateOsvHuntgroup

This method sets the agent status of a hunt group extension.

Table: SetAgentStateOsvHuntgroup

SetAgentStateOsvHuntgroup		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
AgentState	<i>AgentStateOsvHuntgroupEnum</i>	

SetAgentStateOsvHuntgroup		
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>ISetAgentStateOsvHuntgroupDataX</i>		
<i>Events</i>		
<i>OnSetAgentStateOsvHuntgroupEvent</i>		

2.3.2.62 SetForwarding

This method sets the forwarding state of the logged-on extension.

Table: SetForwarding

SetForwarding		
<i>Parameter</i>		
Destination	<i>String</i>	Forwarding destination
Action	<i>SetForwardingActionEnum</i>	Forwarding action
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>ISetForwardingDataX</i>		
<i>Events</i>		
<i>OnSetForwardingEvent</i>		

2.3.2.63 SingleStepTransferCall

This method transfers a call without consultation (blind transfer) to another extension.

Table: SingleStepTransferCall

SingleStepTransferCall		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallId	<i>String</i>	ID of call
Destination	<i>String</i>	Transfer destination
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>ISingleStepTransferCallDataX</i>		
<i>Events</i>		
<i>OnSingleStepTransferCallEvent, OnPhoneStateChangeInfoEvent, OnTransferCompletedEvent</i>		
<i>Comment</i>		
If an empty string is passed to each of the Extension and CallId parameters, the current call is forwarded. Because the messages may be incomplete at the end of this command, depending on the system, it is recommended to implement this command as a ConsultCall-TransferCall sequence.		

2.3.2.64 SnapshotDevice

This method delivers the current device status of an extension.

Table: SnapshotDevice

SnapshotDevice		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>ISnapshotDeviceDataX</i>		
<i>Events</i>		
OnSnapshotDeviceEvent		

2.3.2.65 SnapshotDeviceAdvanced

This method delivers the current connection state of an extension.

Table: SnapshotDeviceAdvanced

SnapshotDeviceAdvanced		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>ISnapshotDeviceAdvancedDataX</i>		
<i>Events</i>		
OnSnapshotDeviceAdvancedEvent		

2.3.2.66 StartMonitorDevice

This method starts the monitoring of the busy lamp field state of an extension.

Table: StartMonitorDevice

StartMonitorDevice		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
UserData	<i>String</i>	Additional user data
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IStartMonitorDeviceDataX</i>		
<i>Events</i>		
OnStartMonitorDeviceEvent		

2.3.2.67 StopMonitorDevice

This method ends the monitoring of the busy lamp field state of an extension.

Table: StopMonitorDevice

StopMonitorDevice		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>IStopMonitorDeviceDataX</i>		
<i>Events</i>		
<i>OnStopMonitorDeviceEvent</i>		

2.3.2.68 TransferCall

This method transfers a call on hold to another extension from a consultation call.

Table: TransferCall

TransferCall		
<i>Parameter</i>		
Extension	<i>String</i>	Extension
CallId	<i>String</i>	ID of call
CallIdOnHold	<i>String</i>	ID of call on hold
Sync _{opt}	<i>Boolean</i>	Synchronous execution [false]
<i>Return</i>		
<i>ITransferCallDataX</i>		
<i>Events</i>		
<i>OnTransferCallEvent, OnPhoneStateChangeInfoEvent, OnTransferCompletedInfoEvent</i>		
<i>Comment</i>		
If an empty string is passed to each of the Extension, CallId and CallIdOnHold parameters, the call currently on hold is forwarded.		

2.3.3 Events

2.3.3.1 OnAcdAgentStateChangeInfoEvent

This event shows a change in the routing state of the agent.

This event can only occur in a nonstandard OSCC-E-V2 configuration when using the HPPC connector.

Table: OnAcdAgentStateChangeInfoEvent

OnAcdAgentStateChangeInfoEvent		
Parameter		
EventData	<i>IAcdAgentStateChangeInfoDataX</i>	Result data

2.3.3.2 OnAcdAvailableEvent

This event shows the result of an asynchronous call of the **AcdAvailable** method.

Table: OnAcdAvailableEvent

OnAcdAvailableEvent		
Parameter		
EventData	<i>IAcdAvailableDataX</i>	Result data

2.3.3.3 OnAcdCheckPasswordEvent

This event shows the result of an asynchronous call of the **AcdCheckPassword** method.

Table: OnAcdCheckPasswordEvent

OnAcdCheckPasswordEvent		
Parameter		
EventData	<i>IAcdCheckPasswordDataX</i>	Result data

2.3.3.4 OnAcdContactStateChangeInfoEvent

This event shows a status change in an ACD contact (OSCC V8 R1 or higher).

Table: OnAcdContactStateChangeInfoEvent

OnAcdContactStateChangeInfoEvent		
Parameter		
EventData	<i>IAcdContactStateChangeInfoDataX</i>	Result data

2.3.3.5 OnAcdContactStateInfoEvent

This event shows detailed information on the state of an ACD contact after a change or an action (OSCC V8 R1 or higher).

Table: OnAcdContactStateInfoEvent

OnAcdContactStateInfoEvent		
Parameter		
EventData	<i>IAcdContactStateInfoDataX</i>	Result data

2.3.3.6 OnAcdExitPostProcessingStateEvent

This event shows the result of an asynchronous call of the **AcdExitPostProcessingState** method (OSCC V8 R0 or higher).

Table: OnAcdExitPostProcessingStateEvent

OnAcdExitPostProcessingStateEvent		
<i>Parameter</i>		
EventData	<i>IAcdExitPostProcessingStateDataX</i>	Result data

2.3.3.7 OnAcdGetBusinessUnitForExtensionEvent

This event shows the result of an asynchronous call of the **AcdGetBusinessUnitForExtension** method.

Table: OnAcdGetBusinessUnitForExtensionEvent

OnAcdGetBusinessUnitForExtensionEvent		
<i>Parameter</i>		
EventData	<i>IAcdGetBusinessUnitForExtensionDataX</i>	Result data

2.3.3.8 OnAcdGetBusinessUnitForUserIdEvent

This event shows the result of an asynchronous call of the **AcdGetBusinessUnitForUserId** method.

Table: OnAcdGetBusinessUnitForUserIdEvent

OnAcdGetBusinessUnitForUserIdEvent		
<i>Parameter</i>		
EventData	<i>IAcdGetBusinessUnitForUserIdDataX</i>	Result data

2.3.3.9 OnAcdGetBusinessUnitsEvent

This event shows the result of an asynchronous call of the **AcdGetBusinessUnits** method.

Table: OnAcdGetBusinessUnitsEvent

OnAcdGetBusinessUnitsEvent		
<i>Parameter</i>		
EventData	<i>IAcdGetBusinessUnitsDataX</i>	Result data

2.3.3.10 OnAcdGetCallInformationEvent

This event shows the result of an asynchronous call of the **AcdGetCallInformation** method.

Table: OnAcdGetCallInformationEvent

OnAcdGetCallInformationEvent		
<i>Parameter</i>		
EventData	<i>IAcdGetCallInformationDataX</i>	Result data

2.3.3.11 OnAcdGetPostProcessingReasonsEvent

This event shows the result of an asynchronous call of the **AcdGetPostProcessingReasons** method (OSCC V8 R0 or higher).

Table: OnAcdGetPostProcessingReasonsEvent

OnAcdGetPostProcessingReasonsEvent		
Parameter		
EventData	<i>IAcdGetPostProcessingReasonsDataX</i>	Result data

2.3.3.12 OnAcdGetPostProcessingStateEvent

This event shows the result of an asynchronous call of the **AcdGetPostProcessingState** method (OSCC V8 R0 or higher).

Table: OnAcdGetPostProcessingStateEvent

OnAcdGetPostProcessingStateEvent		
Parameter		
EventData	<i>IAcdGetPostProcessingStateDataX</i>	Result data

2.3.3.13 OnAcdGetQueuesEvent

This event shows the result of an asynchronous call of the **AcdGetQueue** method.

Table: OnAcdGetQueuesEvent

OnAcdGetQueuesEvent		
Parameter		
EventData	<i>IAcdGetQueuesDataX</i>	Result data

2.3.3.14 OnAcdGetTeamsEvent

This event shows the result of an asynchronous call of the **AcdGetTeams** method.

Table: OnAcdGetTeamsEvent

OnAcdGetTeamsEvent		
Parameter		
EventData	<i>IAcdGetTeamsDataX</i>	Result data

2.3.3.15 OnAcdGetUnavailableReasonsEvent

This event shows the result of an asynchronous call of the **AcdGetUnavailableReason** method.

Table: OnAcdGetUnavailableReasonsEvent

OnAcdGetUnavailableReasonsEvent		
Parameter		
EventData	<i>IAcdGetUnavailableReasonsDataX</i>	Result data

2.3.3.16 OnAcdGetUserIdForUserNameEvent

This event shows the result of an asynchronous call of the **AcdGetUserIdForUserName** method.

Table: OnAcdGetUserIdForUserNameEvent

OnAcdGetUserIdForUserNameEvent		
Parameter		
EventData	<i>IAcdGetUserIdForUserNameDataX</i>	Result data

2.3.3.17 OnAcdGetUsersEvent

This event shows the result of an asynchronous call of the **AcdGetUsers** method.

Table: OnAcdGetUsersEvent

OnAcdGetUsersEvent		
Parameter		
EventData	<i>IAcdGetUsersDataX</i>	Result data

2.3.3.18 OnAcdGetUserStateEvent

This event shows the result of an asynchronous call of the **AcdGetUserState** method.

Table: OnAcdGetUserStateEvent

OnAcdGetUserStateEvent		
Parameter		
EventData	<i>IAcdGetUserStateDataX</i>	Result data

2.3.3.19 OnAcdGetWorkReasonsEvent

This event shows the result of an asynchronous call of the **AcdGetWorkReasons** method.

Table: OnAcdGetWorkReasonsEvent

OnAcdGetWorkReasonsEvent		
Parameter		
EventData	<i>IAcdGetWorkReasonsDataX</i>	Result data

2.3.3.20 OnAcdGetWrapupReasonsEvent

This event shows the result of an asynchronous call of the **AcdGetWrapupReasons** method.

Table: OnAcdGetWrapupReasonsEvent

OnAcdGetWrapupReasonsEvent		
Parameter		
EventData	<i>IAcdGetWrapupReasonsDataX</i>	Result data

2.3.3.21 OnAcdLogoffEvent

This event shows the result of an asynchronous call of the **AcdLogoff** method.

Table: OnAcdLogoffEvent

OnAcdLogoffEvent		
Parameter		
EventData	<i>IAcdLogoffDataX</i>	Result data

2.3.3.22 OnAcdLogonEvent

This event shows the result of an asynchronous call of the **AcdLogon** method.

Table: OnAcdLogonEvent

OnAcdLogonEvent		
Parameter		
EventData	<i>IAcdLogonDataX</i>	Result data

2.3.3.23 OnAcdRegisterForFunctionsEvent

This event shows the result of an asynchronous call of the **AcdRegisterForFunctions** method.

Table: OnAcdRegisterForFunctionsEvent

OnAcdRegisterForFunctionsEvent		
Parameter		
EventData	<i>IAcdRegisterForFunctionsDataX</i>	Result data

2.3.3.24 OnAcdRegisterForGlobalEventsEvent

This event shows the result of an asynchronous call of the **AcdRegisterForGlobalEvents** method.

Table: OnAcdRegisterForGlobalEventsEvent

OnAcdRegisterForGlobalEventsEvent		
Parameter		
EventData	<i>IAcdRegisterForGlobalEventsDataX</i>	Result data

2.3.3.25 OnAcdRegisterForNodeEventsEvent

This event shows the result of an asynchronous call of the **AcdRegisterForNodeEvents** method.

Table: OnAcdRegisterForNodeEventsEvent

OnAcdRegisterForNodeEventsEvent		
Parameter		
EventData	<i>IAcdRegisterForNodeEventsDataX</i>	Result data

2.3.3.26 OnAcdSaveWrapupDataEvent

This event shows the result of an asynchronous call of the **AcdSaveWrapupData** method.

Table: OnAcdWrapupDataEvent

OnAcdSaveWrapupDataEvent		
Parameter		
EventData	<i>IAcdSaveWrapupDataX</i>	Result data

2.3.3.27 OnAcdSetPostProcessingStateEvent

This event shows the result of an asynchronous call of the **AcdSetPostProcessingState** method (OSCC V8 R0 or higher).

Table: OnAcdSetPostProcessingStateEvent

OnAcdSetPostProcessingStateEvent		
Parameter		
EventData	<i>IAcdSetPostProcessingStateDataX</i>	Result data

2.3.3.28 OnAcdSetPrimaryContactEvent

This event shows the result of an asynchronous call of the **AcdSetPrimaryContact** method (OSCC V8 R1 or higher).

Table: OnAcdSetPrimaryContactEvent

OnAcdSetPrimaryContactEvent		
Parameter		
EventData	<i>IAcdSetPrimaryContactDataX</i>	Result data

2.3.3.29 OnAcdUnavailableEvent

This event shows the result of an asynchronous call of the **AcdUnavailable** method.

Table: OnAcdUnavailableEvent

OnAcdUnavailableEvent		
Parameter		
EventData	<i>IAcdUnavailableDataX</i>	Result data

2.3.3.30 OnAcdUnregisterForFunctionsEvent

This event shows the result of an asynchronous call of the **AcdUnregisterForFunctions** method.

Table: OnAcdUnregisterForFunctionsEvent

OnAcdUnregisterForFunctionsEvent		
Parameter		
EventData	<i>IAcdUnregisterForFunctionsDataX</i>	Result data

2.3.3.31 OnAcdUnregisterForGlobalEventsEvent

This event shows the result of an asynchronous call of the **AcdUnregisterForGlobalEvents** method.

Table: OnAcdUnregisterForGlobalEventsEvent

OnAcdUnregisterForGlobalEventsEvent		
<i>Parameter</i>		
EventData	<i>IAcdUnregisterForGlobalEventsDataX</i>	Result data

2.3.3.32 OnAcdUnregisterForNodeEventsEvent

This event shows the result of an asynchronous call of the **AcdUnregisterForNodeEvents** method.

Table: OnAcdUnregisterForNodeEventsEvent

OnAcdUnregisterForNodeEventsEvent		
<i>Parameter</i>		
EventData	<i>IAcdUnregisterForNodeEventsDataX</i>	Result data

2.3.3.33 OnAcdUserStateChangeInfoEvent

This event shows a change in the ACD user state.

Table: OnAcdUserStateChangeInfoEvent

OnAcdUserStateChangeInfoEvent		
<i>Parameter</i>		
EventData	<i>IAcdUserStateChangeInfoDataX</i>	Result data

2.3.3.34 OnAcdVoiceRingingInfoEvent

This event shows an incoming ACD call.

Table: OnAcdVoiceRingingInfoEvent

OnAcdVoiceRingingInfoEvent		
<i>Parameter</i>		
EventData	<i>IAcdVoiceRingingInfoDataX</i>	Result data

2.3.3.35 OnAcdWorkEvent

This event shows the result of an asynchronous call of the **AcdWork** method.

Table: OnAcdWorkEvent

OnAcdWorkEvent		
<i>Parameter</i>		
EventData	<i>IAcdWorkDataX</i>	Result data

2.3.3.36 OnAgentAvailableEvent

This event shows the result of an asynchronous call of the **AgentAvailable** method.

Table: OnAgentAvailableEvent

OnAgentAvailableEvent		
<i>Parameter</i>		
EventData	<i>IAgentAvailableDataX</i>	Result data

2.3.3.37 OnAgentLogoffEvent

This event shows the result of an asynchronous call of the **AgentLogoff** method.

Table: OnAgentLogoffEvent

OnAgentLogoffEvent		
<i>Parameter</i>		
EventData	<i>IAgentLogoffDataX</i>	Result data

2.3.3.38 OnAgentLogonEvent

This event shows the result of an asynchronous call of the **AgentLogon** method.

Table: OnAgentLogonEvent

OnAgentLogonEvent		
<i>Parameter</i>		
EventData	<i>IAgentLogonDataX</i>	Result data

2.3.3.39 OnAgentStateChangeInfoEvent

This event shows a change in the agent status of a phone number.

Table: OnAgentStateChangeInfoEvent

OnAgentStateChangeInfoEvent		
<i>Parameter</i>		
EventData	<i>IAgentStateChangeInfoDataX</i>	Result data

2.3.3.40 OnAgentUnavailableEvent

This event shows the result of an asynchronous call of the **AgentUnavailab** method.

Table: OnAgentUnavailableEvent

OnAgentUnavailableEvent		
<i>Parameter</i>		
EventData	<i>IAgentUnavailableDataX</i>	Result data

2.3.3.41 OnAgentWorkEvent

This event shows the result of an asynchronous call of the **AgentWork** method.

Table: OnAgentWorkEvent

OnAgentWorkEvent		
Parameter		
EventData	<i>IAgentWorkDataX</i>	Result data

2.3.3.42 OnAlternateCallEvent

This event shows the result of an asynchronous call of the **AlternateCall** method.

Table: OnAlternateCallEvent

OnAlternateCallEvent		
Parameter		
EventData	<i>IAlternateCallDataX</i>	Result data

2.3.3.43 OnAnswerCallEvent

This event shows the result of an asynchronous call of the **AnswerCall** method.

Table: OnAnswerCallEvent

OnAnswerCallEvent		
Parameter		
EventData	<i>IAnswerCallDataX</i>	Result data

2.3.3.44 OnCallbackInfoEvent

This event shows the placement of a desired callback.

Table: OnCallbackInfoEvent

OnCallbackInfoEvent		
Parameter		
EventData	<i>ICallbackInfoDataX</i>	Result data

2.3.3.45 OnCallDisconnectOnHeldInfoEvent

This event shows the end of a call.

Table: OnCallDisconnectOnHeldInfoEvent

OnCallDisconnectOnHeldInfoEvent		
Parameter		
EventData	<i>ICallDisconnectOnHeldInfoDataX</i>	Result data

2.3.3.46 OnCallEstablishedInfoEvent

This event shows the acceptance of a call.

Table: OnCallEstablishedInfoEvent

OnCallEstablishedInfoEvent		
Parameter		
EventData	<i>ICallEstablishedInfoDataX</i>	Result data

2.3.3.47 OnCallPickupInfoEvent

This event shows the acceptance of a call without signaling (pickup).

Table: OnCallPickupInfoEvent

OnCallPickupInfoEvent		
Parameter		
EventData	<i>ICallPickupInfoDataX</i>	Result data

2.3.3.48 OnCheckMonitoringDeviceEvent

This event shows the result of an asynchronous call of the **CheckMonitoringDevice** method.

Table: OnCheckMonitoringEvent

OnCheckMonitoringDeviceEvent		
Parameter		
EventData	<i>ICheckMonitoringDeviceDataX</i>	Result data

2.3.3.49 OnClearConnectionEvent

This event shows the result of an asynchronous call of the **ClearConnection** method.

Table: OnClearConnectionEvent

OnClearCallEvent		
Parameter		
EventData	<i>IClearConnectionDataX</i>	Result data

2.3.3.50 OnConferenceCallEvent

This event shows the result of an asynchronous call of the **ConferenceCall** method.

Table: OnConferenceCallEvent

OnConferenceCallEvent		
Parameter		
EventData	<i>IConferenceCallDataX</i>	Result data

2.3.3.51 OnConferenceInfoEvent

This event shows a change in the conference state.

Table: OnConferenceInfoEvent

OnConferenceInfoEvent		
Parameter		
EventData	<i>IConferenceInfoDataX</i>	Result data

2.3.3.52 OnConsultCallEvent

This event shows the result of an asynchronous call of the **ConsultCall** method.

Table: OnConsultCallEvent

OnConsultCallEvent		
Parameter		
EventData	<i>IConsultCallDataX</i>	Result data

2.3.3.53 OnCreateCallbackEvent

This event shows the result of an asynchronous call of the **CreateCallback** method.

Table: OnCreateCallbackEvent

OnCreateCallbackEvent		
Parameter		
EventData	<i>ICreateCallbackDataX</i>	Result data

2.3.3.54 OnDeflectCallEvent

This event shows the result of an asynchronous call of the **DeflectCall** method.

Table: OnDeflectCallEvent

OnDeflectCallEvent		
Parameter		
EventData	<i>IDeflectCallDataX</i>	Result data

2.3.3.55 OnDeleteCallbackEvent

This event shows the result of an asynchronous call of the **DeleteCallback** method.

Table: OnDeleteCallbackEvent

OnDeleteCallbackEvent		
Parameter		
EventData	<i>IDeleteCallbackDataX</i>	Result data

2.3.3.56 OnDialDigitsEvent

This event shows the result of an asynchronous call of the **DialDigits** method.

Table: OnDialDigitsEvent

OnDialDigitsEvent		
Parameter		
EventData	<i>IDialDigitsDataX</i>	Result data

2.3.3.57 OnErrorEvent

This event shows an error message due to a method call.

Table: OnErrorEvent

OnErrorEvent.		
Parameter		
Errorcode	<i>ErrorEnum</i>	Error code

2.3.3.58 OnFailoverInfoEvent

This event shows whether, in the case of geographically separated OSV instances, an error led to a node change.

Table: OnFailoverInfoEvent

OnFailoverInfoEvent		
Parameter		
EventData	<i>IFailoverInfoDataX</i>	Result data

2.3.3.59 OnGetAgentStateEvent

This event shows the result of an asynchronous call of the **GetAgentState** method.

Table: OnGetAgentStateEvent

OnGetAgentStateEvent		
Parameter		
EventData	<i>IGetAgentStateDataX</i>	Result data

2.3.3.60 OnGetAgentStateOsvHuntgroupEvent

This event shows the result of an asynchronous call of the **GetAgentStateOsvHuntgroup** method.

Table: OnGetAgentStateOsvHuntgroupEvent

OnGetAgentStateOsvHuntgroupEvent		
Parameter		
EventData	<i>IGetAgentStateOsvHuntgroupDataX</i>	Result data

2.3.3.61 OnGetDoNotDisturbEvent

This event shows the result of an asynchronous call of the **GetDoNotDisturb** method.

Table: OnGetDoNotDisturbEvent

OnGetDoNotDisturbEvent		
Parameter		
EventData	<i>IGetDoNotDisturbDataX</i>	Result data

2.3.3.62 OnGetForwardingEvent

This event shows the result of an asynchronous call of the **GetForwarding** method.

Table: OnGetForwardingEvent

OnGetForwardingEvent		
Parameter		
EventData	<i>IGetForwardingDataX</i>	Result data

2.3.3.63 OnHoldCallEvent

This event shows the result of an asynchronous call of the **HoldCall** method.

Table: OnHoldCallEvent

OnHoldCallEvent		
Parameter		
EventData	<i>IHoldCallDataX</i>	Result data

2.3.3.64 OnLoggingEvent

This event shows a current log message.

Table: OnLoggingEvent

OnLoggingEvent		
Parameter		
LogSession	<i>String</i>	Session
LogLevel	<i>LogLevelEnum</i>	Log level
Msg	<i>String</i>	Log message

2.3.3.65 OnMakeCallEvent

This event shows the result of an asynchronous call of the **MakeCall** method.

Table: OnMakeCallEvent

OnMakeCallEvent		
Parameter		
EventData	<i>IMakeCallDataX</i>	Result data

2.3.3.66 OnPhoneCallStateChangeInfoEvent

This event shows a change in the current calls at an extension (OpenScape Voice).

Table: OnPhoneCallStateChangeInfoEvent

OnPhoneCallStateChangeInfoEvent		
Parameter		
EventData	<i>IPhoneCallStateChangeInfoDataX</i>	Result data

2.3.3.67 OnPhoneClearedInfoEvent

This event shows that a call at an extension has been disconnected.

Table: OnPhoneClearedInfoEvent

OnPhoneClearedInfoEvent		
Parameter		
EventData	<i>IPhoneClearedInfoDataX</i>	Result data

2.3.3.68 OnPhoneDialingInfoEvent

This event shows that an outgoing calls was initiated from an extension.

Table: OnPhoneDialingInfoEvent

OnPhoneDialingInfoEvent		
Parameter		
EventData	<i>IPhoneDialingInfoDataX</i>	Result data

2.3.3.69 OnPhoneRingingInfoEvent

This event shows that a call arrived at an extension.

Table: OnPhoneRingingInfoEvent

OnPhoneRingingInfoEvent		
Parameter		
EventData	<i>IPhoneRingingInfoDataX</i>	Result data

2.3.3.70 OnPhoneStateBusyInfoEvent

This event shows details on the busy state of an extension.

Table: OnPhoneStateBusyInfoEvent

OnPhoneStateBusyInfoEvent		
Parameter		
EventData	<i>IPhoneStateBusyInfoDataX</i>	Result data

2.3.3.71 OnPhoneStateChangeInfoEvent

This event shows that the call state of the logged-on extension has changed.

Table: OnPhoneStateChangeInfoEvent

OnPhoneStateChangeInfoEvent		
Parameter		

CTI
 "CocCtiConnector" class

OnPhoneStateChangeInfoEvent		
EventData	<i>IPhoneStateChangeInfoDataX</i>	Result data

2.3.3.72 OnReconnectCallEvent

This event shows the result of an asynchronous call of the **ReconnectCall** method.

Table: OnReconnectCallEvent

OnReconnectCallEvent		
<i>Parameter</i>		
EventData	<i>IReconnectCallDataX</i>	Result data

2.3.3.73 OnRetrieveCallEvent

This event shows the result of an asynchronous call of the **RetrieveCall** method.

Table: OnRetrieveCallEvent

OnRetrieveCallEvent		
<i>Parameter</i>		
EventData	<i>IRetrieveCallDataX</i>	Result data

2.3.3.74 OnSendDTMFEvent

This event shows the result of an asynchronous call of the **SendDTMF** method.

Table: OnSendDTMFEvent

OnSendDTMFEvent		
<i>Parameter</i>		
EventData	<i>ISendDTMFDataX</i>	Result data

2.3.3.75 OnServerConnectionStateChangeEvent

This event shows a change in the connection state to the OSCC-E server.

Table: OnServerConnectionStateInfoEvent

OnServerConnectionStateInfoEvent		
<i>Parameter</i>		
EventData	<i>IServerConnectionStateChangeDataX</i>	Result data

2.3.3.76 OnSetAgentStateOsvHuntgroupEvent

This event shows the result of an asynchronous call of the **SetAgentStateOsvHuntgroup** method.

Table: OnSetAgentStateOsvHuntgroupEvent

OnSetAgentStateOsvHuntgroupEvent		
<i>Parameter</i>		
EventData	<i>ISetAgentStateOsvHuntgroupDataX</i>	Result data

2.3.3.77 OnSetForwardingEvent

This event shows the result of an asynchronous call of the **SetForwarding** method.

Table: OnSetForwardingEvent

OnSetForwardingEvent		
<i>Parameter</i>		
EventData	<i>ISetForwardingDataX</i>	Result data

2.3.3.78 OnSingleStepTransferCallEvent

This event shows the result of an asynchronous call of the **SingleStepTransferCall** method.

Table: OnSingleStepTransferCallEvent

OnSingleStepTransferEvent		
<i>Parameter</i>		
EventData	<i>ISingleStepTransferCallDataX</i>	Result data

2.3.3.79 OnSiteStateInfoEvent

This event shows the state of a node.

Table: OnSiteStateInfoEvent

OnSiteStateInfoEvent		
<i>Parameter</i>		
EventData	<i>ISiteStateInfoDataX</i>	Node type

2.3.3.80 OnSnapshotDeviceAdvancedEvent

This event shows the result of an asynchronous call of the **SnapshotDeviceAdvanced** method.

Table: OnSnapshotDeviceAdvancedEvent

OnSnapshotDeviceAdvancedEvent		
<i>Parameter</i>		
EventData	<i>ISnapshotDeviceAdvancedDataX</i>	Result data

2.3.3.81 OnSnapshotDeviceEvent

This event shows the result of an asynchronous call of the **SnapshotDevice** method.

Table: OnSnapshotDeviceEvent

OnSnapshotDeviceEvent		
<i>Parameter</i>		
EventData	<i>ISnapshotDeviceDataX</i>	Result data

2.3.3.82 OnStartMonitorDeviceEvent

This event shows the result of an asynchronous call of the **StartMonitorDevice** method.

Table: OnStartMonitorDeviceEvent

OnStartMonitorDeviceEvent		
Parameter		
EventData	<i>IStartMonitorDeviceDataX</i>	Result data

2.3.3.83 OnStopMonitorDeviceEvent

This event shows the result of an asynchronous call of the **StopMonitorDevice** method.

Table: OnStopMonitorDeviceEvent

OnStopMonitorDeviceEvent		
Parameter		
EventData	<i>IStopMonitorDeviceDataX</i>	Result data

2.3.3.84 OnTransferCallEvent

This event shows the result of an asynchronous call of the **TransferCall** method.

Table: OnTransferCallEvent

OnTransferCallEvent		
Parameter		
EventData	<i>ITransferCallDataX</i>	Result data

2.3.3.85 OnTransferCompletedInfoEvent

This event shows the result of forwarding through a call of **TransferCall** or **SingleStepTransferCall**.

Table: OnTransferCompletedInfoEvent

OnTransferCompletedInfoEvent		
Parameter		
EventData	<i>ITransferCompletedInfoDataX</i>	Result data

2.3.3.86 OnTransferredInfoEvent

This event shows an incoming forwarded call.

Table: OnTransferredInfoEvent

OnTransferredInfoEvent		
Parameter		
EventData	<i>ITransferredInfoDataX</i>	Result data

2.4 Other classes

2.4.1 AcdContactDataListParamX

The **AcdContactDataListParamX** class makes a list of call data available that can be passed as a parameter.

2.4.1.1 Properties

ContactDataList

This property contains the list of call data.

ContactDataList	
<i>IAcdContactDataListX</i>	

2.4.1.2 Methods

Add

This method adds a key value pair to the list.

Table: Add

Add		
<i>Parameter</i>		
Key	<i>String</i>	Key term
Value	<i>String</i>	Value

2.4.2 AcdFunctionTypeListParamX

The **AcdTypeListParamX** class makes a list of function types available that can be passed as a parameter.

2.4.2.1 Properties

FunctionTypeList

This property contains the list of function types.

FunctionTypeList	
<i>IAcdFunctionTypeListX</i>	

2.4.2.2 Methods

Add

This method adds a function type to the list.

Table: Add

Add		
<i>Parameter</i>		
FunctionType	<i>AcdFunctionTypeEnum</i>	Function type

2.4.3 AcdWrapupReasonKeyDataListParamX

The **AcdWrapupReasonKeyDataListParamX** class makes a list of wrapup reasons available that can be passed as a parameter.

2.4.3.1 Properties

WrapupReasonKeyList

This property contains the list of keys for the wrapup reasons.

WrapupReasonKeyList	
<i>I</i> AcdWrapupReasonKeyDataListX	

2.4.3.2 Methods

Add

This method adds the key of a wrapup reason to the list.

Table: Add

Add		
<i>Parameter</i>		
WrapupReasonKey	<i>Integer</i>	Key of the wrapup reason

2.5 Interfaces

2.5.1 IAcidAgentStateChangeInfoDataX

This interface makes the result data for the **OnAcidAgentStateChangeInfoEvent** available.

2.5.1.1 Properties

AcidReasonKey

This property contains the key for an ACD state reason.

AcidReasonKey	
<i>Integer</i>	

AgentId

This property contains the agent ID.

AgentId	
<i>String</i>	

AgentServicePermitted

This property contains the currently possible agent routing control functions.

AgentServicePermitted	
<i>IUserAgentServPermFlagsX</i>	

AgentState

This property contains the agent routing state.

AgentState	
<i>UserAgentStatesEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

2.5.2 IAcdAvailableDataX

This interface makes the result data for the **AcdAvailable** method and the **OnAcdAvailableEvent** available.

2.5.2.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

MediaType

This property contains the media type.

MediaType	
<i>AcdMediaTypeEnum</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

UserId

This property contains the user ID.

UserId	
<i>String</i>	

2.5.3 IAcdBusinessUnitsDataX

This interface makes business unit data available.

2.5.3.1 Properties

BusinessUnitKey

This property contains the key of the business unit.

BusinessUnitKey	
<i>Integer</i>	

BusinessUnitName

This property contains the name of the business unit.

BusinessUnitName	
<i>String</i>	

Description

This property contains the description of the business unit.

Description	
<i>String</i>	

2.5.4 IAcdBusinessUnitsListX

This interface makes a list of IBusinessUnitsDataX objects available.

2.5.4.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.4.2 Methods

get_Item

This methods returns the IPartyDataX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
IBusinessUnitsDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.5 IAcdCheckPasswordDataX

This interface makes the result data for the **AcdCheckPassword** method and the **OnAcdCheckPasswordEvent** available.

2.5.5.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

MediaType

This property contains the media type.

MediaType	
<i>AcdMediaTypeEnum</i>	

PasswordValid

This property indicates whether the password was recognized as valid.

PasswordValid	
<i>Boolean</i>	

UserData

This property contains the additional user data.

UserData	

UserData	
String	

UserId

This property contains the user ID.

UserId	
String	

2.5.6 IAcdContactDataListX

This interface makes a list of IContactDataX objects available.

2.5.6.1 Properties

Count

This property contains the number of lists.

Count	
Integer	

2.5.6.2 Methods

get_Item

This methods returns the IContactDataX object.

get_Item		
Parameters		
Index	Integer	List index
Return		
IacdContactDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
Return		
IEnumerator		

2.5.7 IAcdContactDataX

This interface makes the contact call data available.

2.5.7.1 Properties

Key

This property contains the key term.

Key	
String	

Value

This property contains the value.

Value	
String	

2.5.8 IAcdContactStateAdvancedDataListX

This interface makes a list of IContactStateAdvancedDataX objects available.

2.5.8.1 Properties

Count

This property contains the number of lists.

Count	
Integer	

2.5.8.2 Methods

get_Item

This method returns the IContactStateAdvancedDataX object.

get_Item		
<i>Parameters</i>		
Index	Integer	List index
<i>Return</i>		
IAcdContactStateAdvancedDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.9 IAcdContactStateAdvancedDataX

This interface makes detailed information on an ACD contact available.

2.5.9.1 Properties

AcdHandling

This property indicates whether a contact is being handled by the ACD.

AcdHandling	
Boolean	

AutoWrapup

This property indicates whether post-processing (automatic wrapup) is active.

AutoWrapup	
<i>Boolean</i>	

AutoWrapupSuspended

This property indicates whether automatic wrapup was suspended for a contact.

AutoWrapupSuspended	
<i>Boolean</i>	

ContactId

This property contains the contact ID.

ContactId	
<i>String</i>	

ContactType

This property contains the contact type.

ContactType	
<i>AcdContactTypeEnum</i>	

HandlingState

This property contains the handling state.

HandlingState	
<i>AcdHandlingStateEnum</i>	

MandatoryWrapup

This property specifies whether wrapup reasons must be entered.

MandatoryWrapup	
<i>Boolean</i>	

ManualPostProcessingReason

This property contains the manually set reason for post-processing.

ManualPostProcessingReason	
<i>Integer</i>	

MediaType

This property contains the media type.

Media Type	
<i>AcdMediaTypeEnum</i>	

PostProcessingPending

This property indicates whether manually set post-processing is pending.

PostProcessingPending	
<i>Boolean</i>	

PrimaryContact

This property indicates whether the contact is the primary contact.

PrimaryContact	
<i>Boolean</i>	

RequeueCount

This property contains the number of requeues.

RequeueCount	
<i>Integer</i>	

2.5.10 IAcdContactStateChangeInfoDataX

This interface makes data on the **OnAcdContactStateChangeEvent** available.

2.5.10.1 Properties

ContactStateAdvancedList

This property contains the list of contact information.

ContactStateAdvancedList	
<i>IAcdContactStateAdvancedDataListX</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

MediaType

This property contains the media type.

MediaType	
<i>AcdMediaTypeEnum</i>	

UserId

This property contains the user ID.

UserId	
<i>String</i>	

2.5.11 IAcidContactStateDataListX

This interface makes a list of IContactStateDataX objects available.

2.5.11.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.11.2 Methods

get_Item

This methods returns the IContactStateDataX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
IAcidContactStateDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.12 IAcdContactStateDataX

This interface makes the state data of an ACD contact available.

2.5.12.1 Properties

AcdHandling

This property indicates whether a contact is being handled by the ACD.

AcdHandling	
<i>Boolean</i>	

AutoWrapup

This property indicates whether post-processing (automatic wrapup) is active.

AutoWrapup	
<i>Boolean</i>	

AutoWrapupSuspended

This property indicates whether automatic wrapup was suspended for a contact.

AutoWrapupSuspended	
<i>Boolean</i>	

ContactId

This property contains the contact ID.

ContactId	
<i>String</i>	

ContactType

This property contains the contact type.

ContactType	
<i>AcdContactTypeEnum</i>	

HandlingState

This property contains the handling state.

HandlingState	
<i>AcdHandlingStateEnum</i>	

MandatoryWrapup

This property specifies whether wrapup reasons must be entered.

MandatoryWrapup	
<i>Boolean</i>	

ManualPostProcessingReason

This property contains the manually set reason for post-processing.

ManualPostProcessingReason	
<i>Integer</i>	

MediaType

This property contains the media type.

MediaType	
<i>AcdMediaTypeEnum</i>	

PostProcessingPending

This property indicates whether manually set post-processing is pending.

PostProcessingPending	
<i>Boolean</i>	

PrimaryContact

This property indicates whether the contact is the primary contact.

PrimaryContact	
<i>Boolean</i>	

RequeueCount

This property contains the number of requeues.

RequeueCount	
<i>Integer</i>	

2.5.13 IAcContactStateInfoDataX

This interface makes the result data for the **OnAcdContactStateInfoEvent** available.

This interface makes the result data for the **OnAcdContactStateInfoEvent** available.

2.5.13.1 Properties

ContactStateDataList

This property contains the list of contact states.

ContactStateDataList	
<i>IAcdContactStateDataListX</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

FunctionId

This property contains detailed information on the node type.

FunctionId	
<i>NodeTypeEnum</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

MediaType

This property contains the media type.

MediaType	
<i>MediaTypeEnum</i>	

NodeNumber

This property contains the node number.

NodeNumber	
<i>String</i>	

PostProcessingEnabled

This property indicates whether post-processing is enabled for this contact.

PostProcessingEnabled	
<i>Boolean</i>	

SetPostProcessingReason

This property indicates whether the reason for post-processing can be changed manually for this contact.

SetPostProcessingReason	
<i>Boolean</i>	

UserId

This property contains the user ID.

UserId	
String	

2.5.14 IAcExitPostProcessingStateDataX

This interface makes the result data for the **AcExitPostProcessingState** method and the **OnAcExitPostProcessingStateEvent** available.

2.5.14.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

UserData

This property contains the additional user data.

UserData	
String	

UserId

This property contains the user ID.

UserId	
String	

2.5.15 IAcdGetBusinessUnitForExtensionDataX

This interface makes the result data for the **AcdGetBusinessUnitForExtension** method and the **OnGetBusinessUnitForExtensionEvent** available.

2.5.15.1 Properties

BusinessUnitKey

This property contains the key of the business unit.

BusinessUnitKey	
<i>Integer</i>	

BusinessUnitName

This property contains the name of the business unit.

BusinessUnitName	
<i>String</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
String	

2.5.16 IAcdGetBusinessUnitForUserIdDataX

This interface makes the result data for the **AcdGetBusinessUnitForUserId** method and the **OnGetBusinessUnitForUserIdEvent** available.

2.5.16.1 Properties

BusinessUnitKey

This property contains the key of the business unit.

BusinessUnitKey	
Integer	

BusinessUnitName

This property contains the name of the business unit.

BusinessUnitName	
String	

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

UserData

This property contains the additional user data.

UserData	
----------	--

UserData	
String	

UserId

This property contains the user ID.

UserId	
String	

2.5.17 IAcidGetBusinessUnitsDataX

This interface makes the result data for the **AcidGetBusinessUnits** method and the **OnGetBusinessUnitsEvent** available.

2.5.17.1 Properties

BusinessUnitsList

This property contains the list of the business unit.

BusinessUnitsList	
IAcidBusinessUnitsListX	

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

UserData

This property contains the additional user data.

UserData	
String	

2.5.18 IAcdGetCallInformationDataX

This interface makes the information data on an ACD voice contact available.

2.5.18.1 Properties

AcdCallId

This property contains the ACD call ID.

AcdCallId	
<i>String</i>	

Caption

This property contains the call description.

Caption	
<i>String</i>	

ContactDataList

This property contains the list of call data (key value pairs).

ContactDataList	
<i>IAcdContactDataListX</i>	

ContactType

This property contains the contact type.

ContactType	
<i>AcdContactTypeEnum</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
String	

PabxCallId

This property contains the call ID.

PabxCallId	
String	

QueueKey

This property contains the key of the queue.

QueueKey	
Integer	

QueueName

This property contains the name of the queue.

QueueName	
String	

UserData

This property contains the additional user data.

UserData	
String	

UserId

This property contains the user ID.

UserId	
String	

2.5.18.2 Methods

ToString

This method returns a string that indicates the current object.

ToString		
Return		

2.5.19 IAcdGetPostProcessingReasonsDataX

This interface makes the result data for the **AcdGetPostProcessingReasons** method and the **OnGetPostProcessingReasonsEvent** available.

2.5.19.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

PostProcessingReasonsList

This property contains the list of post-processing reasons.

PostProcessingReasonsList	
<i>IAcdPostProcessingReasonsListX</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

2.5.20 IAcdGetPostProcessingStateDataX

This interface makes the result data for the **AcdGetPostProcessingState** method and the **OnAcdGetPostProcessingStateEvent** available.

2.5.20.1 Properties

ContactStateDataList

This property contains the list of contact states.

ContactStateDataList	
<i>IAcdContactStateDataListX</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

UserId

This property contains the user ID.

UserId	
<i>String</i>	

2.5.21 IAcdGetQueuesDataX

This interface makes the result data for the **AcdGetQueues** method and the **OnAcdGetQueuesEvent** available.

2.5.21.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
String	

MediaType

This property contains the media type.

MediaType	
MediaTypeEnum	

QueueDataList

This property contains the list of queue data.

QueueDataList	
IAcdQueueDataListX	

UserData

This property contains the additional user data.

UserData	
String	

2.5.21.2 Methods

ToString

This method returns a string that indicates the current object.

ToString		
Return		

2.5.22 IAcdGetTeamsDataX

This interface makes the result data for the **AcdGetTeams** method and the **OnAcdGetTeamsEvent** available.

2.5.22.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

QueueDataList

This property contains the list of queue data.

TeamDataList	
IacdTeamDataListX	

UserData

This property contains the additional user data.

UserData	
String	

2.5.22.2 Methods

ToString

This method returns a string that indicates the current object.

ToString		
Return		

2.5.23 IAcidGetUnavailableReasonsDataX

This interface makes the result data for the **AcidGetUnavailableReasons** method and the **OnAcidGetUnavailableReasonsEvent** available.

2.5.23.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

UnavailableReasonsDataList

This property contains the list of unavailability reasons.

UnavailableReasonsDataList	
IAcdUnavailableReasonsDataListX	

UserData

This property contains the additional user data.

UserData	
String	

2.5.23.2 Methods

ToString

This method returns a string that indicates the current object.

ToString		
Return		

2.5.24 IAcdGetUserIdForUserNameDataX

This interface makes the result data for the **AcdGetUserIdForUserName** method and the **OnAcdGetUserIdForUserNameEvent** available.

2.5.24.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

UserData

This property contains the additional user data.

UserData	
String	

UserId

This property contains the user ID.

UserId	
String	

2.5.25 IAcdGetUsersDataX

This interface makes the result data for the **AcdGetUsers** method and the **OnAcdGetUsersEvent** available.

2.5.25.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

UserData

This property contains the additional user data.

UserData	
String	

UserDataList

This property contains the list of user data.

UserDataList	
IAcdUserDataListX	

2.5.26 IAcdGetUserStateDataX

This interface makes the result data for the **AcdGetUserState** method and the **OnAcdGetUserStateEvent** available.

2.5.26.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

Extension

This property contains the extension.

Extension	
String	

LocationId

This property contains the node name.

LocationId	
String	

LoggedOnMediaTypeIdList

This property contains the list of media types for which the user is logged on.

LoggedOnMediaTypeIdList	
IAcdMediaTypeIdListX	

MediaType

This property contains the media type (in case of a negative reply, otherwise awtNone).

MediaType	
AcdMediaTypeIdEnum	

UserData

This property contains the additional user data.

UserData	
String	

UserId

This property contains the user ID.

UserId	
String	

UserState

This property contains the ACD routing state.

UserState	
UserAgentStatesEnum	

2.5.27 IAcdGetWorkReasonsDataX

This interface makes the result data for the **AcdGetWorkReasons** method and the **OnAcdGetWorkReasonsEvent** available.

2.5.27.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

WorkReasonsDataList

This property contains the list of work reasons.

WrapupReasonsDataList	
<i>IAcdWorkReasonDataListX</i>	

2.5.28 IAcdGetWrapupReasonsDataX

This interface makes the result data for the **AcdGetWrapupReasons** method and the **OnAcdGetWrapupReasonsEvent** available.

2.5.28.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

UserData

This property contains the additional user data.

UserData	
String	

WrapupReasonsDataList

This property contains the list of wrapup reasons.

WrapupReasonsDataList	
IAcdWrapupReasonsDataListX	

2.5.28.2 Methods

ToString

This method returns a string that indicates the current object.

ToString		
Return		

2.5.29 IAcdLogoffDataX

This interface makes the result data for the **AcdLogoff** method and the **OnAcdLogoffEvent** available.

2.5.29.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

MediaType

This property contains the media type.

MediaType	
AcdMediaTypeEnum	

UserData

This property contains the additional user data.

UserData	
String	

UserId

This property contains the user ID.

UserId	
String	

2.5.30 IAcLogonDataX

This interface makes the result data for the **AcLogon** method and the **OnAcLogonEvent** available.

2.5.30.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

MediaType

This property contains the media type.

MediaType	
AcdMediaTypeEnum	

UserData

This property contains the additional user data.

UserData	
String	

UserId

This property contains the user ID.

UserId	
String	

2.5.31 IAcidMediaTypeDataX

This interface makes data of a media type available.

2.5.31.1 Properties

MediaType

This property contains the media type.

MediaType	
AcdMediaTypeEnum	

2.5.32 IAcdMediaTypeListX

This interface makes a list of IAcdMediaTypeDataX objects available.

2.5.32.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.32.2 Methods

get_Item

This methods returns the IAcdMediaTypeDataX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
IAcdMediaTypeDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.33 IAcdPostProcessingReasonsDataX

This interface makes data of a post-processing reason available.

2.5.33.1 Properties

BusinessUnitKey

This property contains the key of the business unit.

BusinessUnitKey	
<i>Integer</i>	

Description

This property contains a description of the post-processing reason.

Description	
<i>String</i>	

ReasonKey

This property contains the key of the post-processing reason.

ReasonKey	
<i>Integer</i>	

ReasonName

This property contains the name of the post-processing reason.

ReasonName	
<i>String</i>	

ReferenceKey

This property contains the reference key of the post-processing reason.

ReferenceKey	
<i>Integer</i>	

2.5.34 IAcPostProcessingReasonsListX

This interface makes a list of IAcPostProcessingReasonsDataX objects available.

2.5.34.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.34.2 Methods

get_Item

This methods returns the IAcPostProcessingReasonsDataX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
IAcPostProcessingReasonsDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.35 IAcdPostProcessingServPermFlagsX

This interface makes the possible actions in the post-processing state available.

2.5.35.1 Properties

Exit

This property indicates whether post-processing can be ended.

Exit	
<i>Boolean</i>	

SetReason

This property indicates whether the reason for post-processing can be changed manually.

SetReason	
<i>Boolean</i>	

2.5.35.2 Methods

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.36 IAcdQueueDataListX

This interface makes a list of IAcdQueueDataX objects available.

2.5.36.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.36.2 Methods

get_Item

This methods returns the IAcdQueueDataX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
IAcdQueueDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.37 IAcdQueueDataX

This interface makes detailed information on a queue available.

2.5.37.1 Properties

BusinessUnitKey

This property contains the key of the business unit.

BusinessUnitKey	
<i>Integer</i>	

Description

This property contains the description of the queue.

Description	
<i>String</i>	

MediaType

This property contains the media type.

MediaType	
<i>AcdMediaTypeEnum</i>	

QueueKey

This property contains the key of the queue.

QueueKey	
<i>Integer</i>	

QueueName

This property contains the queue name.

QueueName	
<i>String</i>	

WrapupReasonKeyDataList

This property contains the list of wrapup reasons belonging to the queue.

WrapupReasonKeyDataList	
<i>IAcdWrapupReasonKeyDataList</i>	

2.5.38 IAcdRegisterForFunctionsDataX

This interface makes the result data for the **AcdRegisterForFunctions** method and the **OnAcdRegisterForFunctionsEvent** available.

2.5.38.1 Properties

Errorcode

This property contains the error code.

Error code	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

FunctionTypeList

This property contains the list of registered, special functions.

FunctionTypeList	
<i>IAcdFunctionTypeListX</i>	

FunctionTypeListStringMask

This property contains a string representation of the list of registered, special functions.

FunctionTypeListStringMask	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

UserId

This property contains the user ID.

UserId	
String	

2.5.39 IAcdRegisterForGlobalEventsDataX

This interface makes the result data for the **AcdRegisterForGlobalEvents** method and the **OnAcdRegisterForGlobalEventsEvent** available.

2.5.39.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

UserData

This property contains the additional user data.

UserData	
String	

2.5.40 IAcdRegisterForNodeEventsDataX

This interface makes the result data for the **AcdRegisterForNodeEvents** method and the **OnAcdRegisterForNodeEventsEvent** available.

2.5.40.1 Properties

AcdRelease

This property contains the ACD release number.

AcdRelease	
Integer	

AcdVersion

The property contains the ACD version number.

AcdVersion	
<i>Integer</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

MediaType

This property contains the media type.

MediaType	
<i>AcdMediaTypeEnum</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

UserId

This property contains the user ID.

UserId	
<i>String</i>	

2.5.41 IAcdSaveWrapupDataX

This interface makes the result data for the **AcdSaveWrapupData** method and the **OnAcdSaveWrapupDataEvent** available.

2.5.41.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

UserId

This property contains the user ID.

UserId	
<i>String</i>	

2.5.41.2 Methods

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
<i>String</i>		

2.5.42 IAcidSetPostProcessingStateDataX

This interface makes the result data for the **AcidSetPostProcessingState** method and the **OnAcidSetPostProcessingStateEvent** available.

2.5.42.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

UserId

This property contains the user ID.

UserId	
<i>String</i>	

2.5.42.2 Methods

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.43 IAcdSetPrimaryContactDataX

This interface makes the result data for the **AcdSetPrimaryContact** method and the **OnAcdSetPrimaryContactEvent** available.

2.5.43.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

UserId

This property contains the user ID.

UserId	
<i>String</i>	

2.5.43.2 Methods

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.44 IAcdTeamDataListX

This interface makes a list of IAcdTeamDataX objects available.

2.5.44.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.44.2 Methods

get_Item

This methods returns the IAcTeamDataX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
IAcTeamDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.45 IAcTeamDataX

This interface makes detailed information on an agent team available.

2.5.45.1 Properties

BusinessUnitKey

This property contains the key of the business unit.

BusinessUnitKey	
<i>Integer</i>	

Description

This property contains a description of the team.

Description	
<i>String</i>	

TeamKey

This property contains the key of the team.

TeamKey	
<i>Integer</i>	

TeamName

This property contains the team name.

TeamName	
String	

2.5.46 IAcdUnavailableDataX

This interface makes the result data for the **AcdUnavailable** method and the **OnAcdUnavailableEvent** available.

2.5.46.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

MediaType

This property contains the media type.

MediaType	
AcdMediaTypeEnum	

UserData

This property contains the additional user data.

UserData	
String	

UserId

This property contains the user ID.

UserId	
String	

2.5.47 IAcdUnavailableReasonDataListX

This interface makes a list of IAcdUnavailableReasonDataX objects available.

2.5.47.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.47.2 Methods

get_Item

This methods returns the IAcdUnavailableReasonDataX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
IAcdUnavailableReasonDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.48 IAcdUnavailableReasonDataX

This interface makes data of an unavailability reason available.

2.5.48.1 Properties

BusinessUnitKey

This property contains the key of the business unit.

BusinessUnitKey	
<i>Integer</i>	

Description

This property contains a description of the unavailability reason.

Description	
<i>String</i>	

ReasonKey

This property contains the key of the unavailability reason.

ReasonKey	
<i>Integer</i>	

ReasonName

This property contains the name of the unavailability reason.

ReasonName	
<i>String</i>	

ReferenceKey

This property contains the reference key of the unavailability reason.

ReferenceKey	
<i>Integer</i>	

2.5.49 IAcdUnregisterForFunctionsDataX

This interface makes the result data for the **AcdUnregisterForFunctions** method and the **OnAcdUnregisterForFunctionsEvent** available.

2.5.49.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

ErrorData

This property contains the error description.

ErrorData	
<i>String</i>	

FunctionTypeList

This property contains the list of registered, special functions.

FunctionTypeList	
<i>IAcdFunctionTypeListX</i>	

FunctionTypeListStringMask

This property contains a string representation of the list of registered, special functions.

FunctionTypeListStringMask	
String	

LocationId

This property contains the node name.

LocationId	
String	

UserData

This property contains the additional user data.

UserData	
String	

UserId

This property contains the user ID.

UserId	
String	

2.5.50 IAcdUnregisterForGlobalEventsDataX

This interface makes the result data for the **AcdUnregisterForGlobalEvents** method and the **OnAcdUnregisterForGlobalEventsEvent** available.

2.5.50.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

UserData

This property contains the additional user data.

UserData	
String	

2.5.51 IAcUnregisterForNodeEventsDataX

This interface makes the result data for the **AcUnregisterForNodeEvents** method and the **OnAcUnregisterForNodeEventsEvent** available.

2.5.51.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

MediaType

This property contains the media type.

MediaType	
AcdMediaTypeEnum	

UserData

This property contains the additional user data.

UserData	
String	

UserId

This property contains the user ID.

UserId	
String	

2.5.52 IAcdUserConfigDataX

The interface makes the data for the ACD user logon available.

2.5.52.1 Properties

BusinessUnitName

This property contains the name of the business unit.

BusinessUnitName	
String	

Password

This property contains the user password.

Password	
String	

UserId

This property contains the ACD user ID.

UserId	
String	

2.5.53 IAcdUserDataX

This interface makes the user data available.

2.5.53.1 Properties

BusinessUnitKey

This property contains the key of the business unit.

BusinessUnitKey	
Integer	

DepartmentKey

This property contains the department key.

DepartmentKey	
<i>Integer</i>	

Firstname

This property contains the first name.

Firstname	
<i>String</i>	

Lastname

This property contains the last name.

Lastname	
<i>String</i>	

MediaTypeList

This property contains the list of media types.

MediaTypeList	
<i>IAcdMediaTypeList</i>	

UserId

This property contains the user ID.

UserId	
<i>String</i>	

UserKey

This property contains the user key.

UserKey	
<i>Integer</i>	

UserName

This property contains the ACD user name.

UserName	
<i>String</i>	

2.5.54 IAcidUserDataListX

This interface makes a list of IAcidUserDataX objects available.

2.5.54.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.54.2 Methods

get_Item

This methods returns the IAcidUserDataX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
IAcidUserDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.55 IAcidUserStateChangeInfoDataX

This interface makes the result data for the **OnAcidUserStateChangeInfoEvent** available.

2.5.55.1 Properties

AcidReasonKey

This property contains the key of the state change reason.

AcidReasonKey	
<i>Integer</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

LocationId

This property contains the node name.

LocationId	
String	

MediaTypeList

This property contains the list of media types for which the user is logged on.

LoggedOnMediaTypeList	
IAcdMediaTypeListX	

UserId

This property contains the user ID.

UserId	
String	

UserServicePermitted

This property contains the options currently available for changing the routing state.

UserServicePermittedFlags	
IUserAgentServPermFlagsX	

UserState

This property contains the ACD routing state.

UserState	
UserAgentStatesEnum	

2.5.56 IAcdVoiceRingingInfoDataX

This interface makes the data for the **OnAcdVoiceRingingInfoEvent** available.

2.5.56.1 Properties

AcdCallId

This property contains the ACD call ID.

AcdCallId	
String	

Caption

This property contains the call description.

Caption	
---------	--

Caption	
<i>String</i>	

ContactDataList

This property contains the list of call data (key value pairs).

ContactDataList	
<i>IAcdContactDataListX</i>	

ContactType

This property contains the contact type.

ContactType	
<i>AcdContactTypeEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

FromPartyAgentKey

This property contains the agent key of the call source.

FromPartyAgentKey	
<i>Integer</i>	

FromPartyDevice

This property contains the phone number of the call source.

FromPartyDevice	
<i>String</i>	

FromPartyPartyType

This property contains the subscriber type of the call source.

FromPartyPartyType	
<i>AcdPartyTypeEnum</i>	

FunctionId

This property contains detailed information on the node type.

FunctionId	
<i>NodeTypeEnum</i>	

LocationId

This property contains the node name.

LocationId	
<i>String</i>	

NodeNumber

This property contains the node number.

NodeNumber	
<i>String</i>	

OriginalFromPartyAgentKey

This property contains the agent key of the original call source.

OriginalFromPartyAgentKey	
<i>Integer</i>	

OriginalFromPartyDevice

This property contains the phone number of the original call source.

OriginalFromPartyDevice	
<i>String</i>	

OriginalFromPartyPartyType

This property contains the subscriber type of the original call source.

OriginalFromPartyPartyType	
<i>AcdPartyTypeEnum</i>	

OriginalToPartyAgentKey

This property contains the agent key of the original call destination.

OriginalToPartyAgentKey	
<i>Integer</i>	

OriginalToPartyDevice

This property contains the phone number of the original call destination.

OriginalToPartyDevice	
<i>String</i>	

OriginalToPartyPartyType

This property contains the subscriber type of the original call destination.

OriginalToPartyPartyType	
<i>AcdPartyTypeEnum</i>	

PabxCallId

This property contains the call ID.

PabxCallId	
<i>String</i>	

Priority

This property contains the call priority.

Priority	
<i>Integer</i>	

QueueKey

This property contains the key of the queue.

QueueKey	
<i>Integer</i>	

QueueName

This property contains the name of the queue.

QueueName	
<i>String</i>	

ToPartyAgentKey

This property contains the agent key of the call destination.

ToPartyAgentKey	
<i>Integer</i>	

ToPartyDevice

This property contains the phone number of the call destination.

ToPartyDevice	
<i>String</i>	

ToPartyPartyType

This property contains the subscriber type of the call destination.

ToPartyPartyType	
<i>AcdPartyTypeEnum</i>	

UserId

This property contains the ACD user ID.

UserId	
String	

2.5.57 IAcdWorkDataX

This interface makes the result data for the **AcdWork** method and the **OnAcdWorkEvent** available.

2.5.57.1 Properties

ErrorCode

This property contains the error code.

ErrorCode	
ErrorEnum	

ErrorData

This property contains the error description.

ErrorData	
String	

LocationId

This property contains the node name.

LocationId	
String	

MediaType

This property contains the media type.

MediaType	
AcdMediaTypeEnum	

UserData

This property contains the additional user data.

UserData	
String	

UserId

This property contains the user ID.

UserId	
String	

2.5.58 IAcWorkReasonDataListX

This interface makes a list of IAcWorkReasonDataX objects available.

2.5.58.1 Properties

Count

This property contains the number of lists.

Count	
Integer	

2.5.58.2 Methods

get_Item

This methods returns the IAcWorkReasonDataX object.

get_Item		
<i>Parameters</i>		
Index	Integer	List index
<i>Return</i>		
IAcWorkReasonDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.59 IAcWorkReasonDataX

This interface makes data of a work reason available.

2.5.59.1 Properties

BusinessUnitKey

This property contains the key of the business unit.

BusinessUnitKey	
Integer	

Description

This property contains a description of the work reason.

Description	
String	

ReasonKey

This property contains the key of the work reason.

ReasonKey	
Integer	

ReasonName

This property contains the name of the work reason.

ReasonName	
String	

ReferenceKey

This property contains the reference key of the work reason.

ReferenceKey	
Integer	

2.5.60 IAcidWrapupReasonDataX

This interface makes a wrapup reason available.

2.5.60.1 Properties

BusinessUnitKey

This property contains the key of the business unit.

BusinessUnitKey	
Integer	

Description

This property contains a description of the wrapup reason.

Description	
String	

ReasonKey

This property contains the key of the wrapup reason.

ReasonKey	
Integer	

ReasonName

This property contains the name of the wrapup reason.

ReasonName	
String	

2.5.61 IAcWrapupReasonKeyDataListX

This interface makes a list of IAcWrapupReasonKeyDataX objects available.

2.5.61.1 Properties

Count

This property contains the number of lists.

Count	
Integer	

2.5.61.2 Methods

get_Item

This methods returns the IAcWrapupReasonKeyDataX object.

get_Item		
<i>Parameters</i>		
Index	Integer	List index
<i>Return</i>		
IAcWrapupReasonKeyDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.62 IAcWrapupReasonKeyDataX

This interface makes a key for a wrapup reason available.

2.5.62.1 Properties

WrapupReasonKey

This property contains the key of the wrapup reason.

WrapupReasonKey	
Integer	

2.5.63 IAcWrapupReasonsDataListX

This interface makes a list of IAcWrapupReasonsDataX objects available.

2.5.63.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.63.2 Methods

get_Item

This methods returns the IAcWrapupReasonsDataX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
IAcWrapupReasonsDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.64 IAgentAvailableDataX

This interface makes the result data for the **AgentAvailable** method and the **OnAgentAvailableEvent** available.

2.5.64.1 Properties

AgentState

This property contains details on the agent state of an extension.

AgentState	
<i>UserAgentStatesEnum</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.64.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.65 IAgentLogoffDataX

This interface makes the result data for the **AgentLogoff** method and the **OnAgentLogoffEvent** available.

2.5.65.1 Properties

AgentState

This property contains details on the agent state of an extension.

AgentState	
<i>UserAgentStatesEnum</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
String	

2.5.65.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
Return		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
Return		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
Return		
String		

2.5.66 IAgentLogonDataX

This interface makes the result data for the **AgentLogon** method and the **OnAgentLogonEvent** available.

2.5.66.1 Properties

AgentState

This property contains details on the agent state of an extension.

AgentState	
UserAgentStatesEnum	

CstaError

This property contains details on the CSTA error.

CstaError	
ICstaErrorX	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.66.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.67 IAgentStateChangeInfoDataX

This interface makes the result data for the **OnAgentStateChangeInfoEvent** available.

2.5.67.1 Properties

AgentId

This property contains the agent ID.

AgentId	
<i>String</i>	

AgentServicePermitted

This property contains the options currently available for changing the routing state.

AgentServicePermittedFlags	
<i>IUserAgentServPermFlagsX</i>	

AgentState

This property contains the agent routing state.

AgentState	
<i>UserAgentStatesEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

2.5.68 IAgentUnavailableDataX

This interface makes the result data for the **AgentUnavailable** method and the **OnAgentUnavailableEvent** available.

2.5.68.1 Properties

AgentState

This property contains details on the agent state of an extension.

AgentState	
<i>UserAgentStatesEnum</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.68.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.69 IAgentWorkDataX

This interface makes the result data for the **AgentWork** method and the **OnAgentWorkEvent** available.

2.5.69.1 Properties

AgentState

This property contains details on the agent state of an extension.

AgentState	
<i>UserAgentStatesEnum</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.69.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.70 IAlternateCallDataX

This interface makes the result data for the **AlternateCall** method and the **OnAlternateCallEvent** available.

2.5.70.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.70.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.71 IAnswerCallDataX

This interface makes the result data for the **AnswerCall** method and the **OnAnswerCallEvent** available.

2.5.71.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.71.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.72 ICallbackInfoDataX

This interface makes the result data for the **OnCallbackInfoEvent** available.

2.5.72.1 Properties

CallbackDevice

This property contains the extension to which the callback is established.

CallbackDevice	
String	

CallId

This property contains the call ID of the initiated callback.

CallId	
String	

Extension

This property contains the extension.

Extension	
String	

SiteId

This property contains the node name.

SiteId	
String	

2.5.73 ICallDisconnectOnHeldInfoDataX

This interface makes the result data for the **OnCallDisconnectOnHeldInfoEvent** available.

2.5.73.1 Properties

CallId

This property contains the call ID of the call on hold.

CallId	
String	

DisconnectingDevice

This property contains the disconnecting extension.

DisconnectingDevice	
String	

Extension

This property contains the extension.

Extension	
String	

2.5.74 ICallEstablishedInfoDataX

This interface makes the information data for the **OnCallEstablishedInfoEvent** available.

2.5.74.1 Properties

AnsweringDevice

This property contains the accepting phone number.

AnsweringDevice	
String	

CallAnswered

This property indicates whether a call was accepted.

CallAnswered	
Boolean	

CallId

This property contains the call ID of the disconnected call.

CallId	
String	

CallingDevice

This property contains the incoming phone number.

CallingDevice	
String	

EventCause

This property contains the event reason for a CSTA command.

EventCause	
EventCauseEnum	

Extension

This property contains the extension.

Extension	
String	

LastRedirectingDevice

This property contains the extension of the last call redirection.

LastRedirectingDevice	
String	

OriginalCalledDevice

This property contains the originally called number.

OriginalCalledDevice	
String	

2.5.75 ICallPickupInfoDataX

This interface makes the information data for the **OnCallPickupInfoEvent** available.

2.5.75.1 Properties

CallId

This property contains the call ID of the call.

CallId	
String	

CallingDevice

This property contains the incoming phone number.

CallingDevice	
String	

Extension

This property contains the extension.

Extension	
String	

LastRedirectingDevice

This property contains the extension of the last call redirection.

LastRedirectingDevice	
String	

OriginalCalledDevice

This property contains the originally called number.

OriginalCalledDevice	
String	

PickupDevice

This property contains the extension that picks up the call.

PickupDevice	
<i>String</i>	

Siteld

This property contains the node name.

Siteld	
<i>String</i>	

2.5.76 ICheckMonitoringDeviceDataX

This interface makes the result data for the **CheckMonitoringDevice** method and the **OnCheckMonitoringDeviceEvent** available.

2.5.76.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

MonitorDeviceHandlingFlag

This property contains details on the type of monitoring of an extension.

MonitorDeviceHandlingFlag	
<i>MonitorDeviceHandlingFlagEnum</i>	

MonitorState

This property contains details on the monitoring state of an extension.

MonitorState	
<i>MonitorStateEnum</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

2.5.76.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.77 IClearConnectionDataX

This interface makes the result data for the **ClearConnection** method and the **OnClearConnectionEvent** available.

2.5.77.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.77.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.78 ICommonConfigDataX

This interface makes general configuration and setting data available.

2.5.78.1 Properties

ApplicationIdentifier

This property contains the application identifier required for the license check.

ApplicationIdentifier	
<i>ApplicationIdentifierEnum</i>	

DefaultUserData

This property contains the default user data that are used for various functions and events.

DefaultUserData	
<i>String</i>	

IPVersion

This property contains the IP protocol version in use.

IPVersion	
<i>IPVersionEnum</i>	

SynchronTimeout

This property contains the timeout value for synchronous method execution in milliseconds.

SynchronTimeout	
<i>Integer</i>	

2.5.79 IConferenceCallDataX

This interface makes the result data for the **ConferenceCall** method and the **OnConferenceCallEvent** available.

2.5.79.1 Properties

CallId

This property contains the call ID of the conference.

CallId	
<i>String</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.79.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.80 IConferenceInfoDataX

This interface makes the result data for the **OnConferenceInfoEvent** available.

2.5.80.1 Properties

CallId

This property contains the call ID of the conference.

CallId	
<i>String</i>	

ConferenceCreator

This property indicates whether a subscriber convened the conference.

ConferenceCreator	
<i>Boolean</i>	

ConferenceEventType

This property contains the event that led to a change in the conference state.

ConferenceEventType	
<i>ConferenceEventTypeEnum</i>	

ConsultCallId

This property contains the call ID with the consulted extension.

ConsultCallId	
<i>String</i>	

EventCause

This property contains the event reason for a CSTA command.

EventCause	
<i>EventCauseEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

HoldCallId

This property contains the call ID with the extension on hold.

HoldCallId	
<i>String</i>	

Lcs

This property contains the connection state.

Lcs	
<i>LocalConnectionStateEnum</i>	

PartyList

This property contains the subscriber list of the conference.

PartyList	
<i>IPartyListX</i>	

ReplacedCallId

This property contains the call ID replaced by the call ID of the conference.

ReplacedCallId	
<i>String</i>	

SilentMonitoring

This property indicates whether silent monitoring is enabled.

SilentMonitoring	
<i>Boolean</i>	

Siteld

This property contains the node name.

Siteld	
String	

2.5.81 IConfigDataX

This interface makes configuration and setting data available.

2.5.81.1 Properties

AcidUserConfigData

This property contains the user data for the ACD logon.

AcidUserConfigData	
AcidUserConfigDataX	

CommonConfigData

This property contains general configuration data.

CommonConfigData	
ICommonConfigDataX	

CtiConnectivityConfigData

This property contains the configuration data for the CTI connection.

CtiConnectivityConfigData	
CtiConnectivityConfigDataX	

LogConfigData

This property contains the settings for the process log for program diagnosis.

LogConfigData	
ILoggingConfigDataX	

2.5.82 IConnectionDataX

This interface makes detailed information on a voice connection available.

2.5.82.1 Properties

ActiveCallId

This property contains the call ID of the connection.

ActiveCallId	
String	

Device

This property contains the extension or the trunk number (for external subscribers).

Device	
<i>String</i>	

DeviceDisplay

This property contains the display of the extension.

DeviceDisplay	
<i>String</i>	

DeviceType

This property contains the extension type.

DeviceType	
<i>DeviceTypeEnum</i>	

OldCallIdList

This property contains the list of former call IDs.

OldCallIdList	
<i>IOldCallIdListX</i>	

Position

This property indicates the position of the extension in the list of connection subscribers.

Position	
<i>Integer</i>	

2.5.83 IConnectionListX

This interface makes a list of IConnectionDataX objects available.

2.5.83.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.83.2 Methods

get_Item

This methods returns the IConnectionDataX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
IConnectionDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.84 IConsultCallDataX

This interface makes the result data for the **ConsultCall** method and the **OnConsultCallEvent** available.

2.5.84.1 Properties

CallId

This property contains the call ID of the consultation call.

CallId	
<i>String</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.84.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.85 ICreateCallbackDataX

This interface makes the result data for the **CreateCallback** method and the **OnCreateCallbackEvent** available.

2.5.85.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Destination

This property contains the callback number.

Destination	
<i>String</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
String	

2.5.85.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
Return		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
Return		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
Return		
String		

2.5.86 ICstaErrorX

This interface makes detailed information on a CSTA error available.

2.5.86.1 Properties

CstaErrorClass

This property contains the CSTA error class value.

CstaErrorClass	
Integer	

CstaErrorClassTxt

This property contains the CSTA error class description.

CstaErrorClassTxt	
String	

CstaErrorValue

This property contains the CSTA error value.

CstaErrorValue	
<i>Integer</i>	

CstaErrorValueTxt

This property contains the CSTA error description.

CstaErrorValueTxt	
<i>String</i>	

2.5.86.2 Methods

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.87 ICtiConnectivityConfigDataX

This interface makes general configuration and setting data for the CTI connection available.

2.5.87.1 Properties

AutoReconnect

This property indicates whether an attempt should automatically be made to reestablish the connection if a connection is lost.

AutoReconnect	
<i>Boolean</i>	

ConnectAndMonitor

This property indicates whether the state of the extension should be monitored after a successful logon.

ConnectAndMonitor	
<i>Boolean</i>	

Extension

This property contains the extension. The E164 format can be used.

Extension	
<i>String</i>	

IpAddress

This property contains the IP address of the Integration Connector.

IpAddress	
<i>String</i>	

Port

This property contains the port number of the Integration Connector.

Port	
<i>Integer</i>	

Siteld

This property contains the node name.

Siteld	
<i>String</i>	

2.5.88 IDeflectCallDataX

This interface makes the result data for the **DeflectCall** method and the **OnDeflectCallEvent** available.

2.5.88.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.88.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.89 IDeleteCallbackDataX

This interface makes the result data for the **DeleteCallback** method and the **OnDeleteCallbackEvent** available.

2.5.89.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.89.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.90 IDialDigitsDataX

This interface makes the result data for the **DialDigits** method and the **OnDialDigitsEvent** available.

2.5.90.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.90.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.91 IExtensionCallDataX

This interface makes information on a connected voice call available.

2.5.91.1 Properties

Active

This property indicates whether the call is active.

Active	
<i>Boolean</i>	

CallId

This property contains the call ID.

CallId	
<i>String</i>	

CallState

This property contains the connection state.

CallState	
<i>CallStateEnum</i>	

Cause

This property contains the event reason for a CSTA command.

Cause	
<i>EventCauseEnum</i>	

Direction

This property contains the connection direction.

Direction	
<i>CallDirectionEnum</i>	

PartyList

This property contains the subscriber list of the conference.

PartyList	
<i>IPartyListX</i>	

RemoteParty

This property contains the remote party.

RemoteParty	
<i>String</i>	

RemotePartyBusy

This property indicates whether the remote party is busy.

RemotePartyBusy	
<i>Boolean</i>	

ReplacedCallId

This property contains the call ID replaced by the call ID of the conference.

ReplacedCallId	
<i>String</i>	

ServicePermittedFlags

This property contains the CTI functions currently available.

ServicePermittedFlags	
<i>IServPermFlagsX</i>	

2.5.92 IExtensionCallListX

This interface makes a list of IExtensionCallDataX objects available.

2.5.92.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.92.2 Methods

get_Item

This methods returns the IExtensionCallDataX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
IExtensionCallDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.93 IFailoverInfoDataX

This interface makes the result data for the **OnFailoverInfoEvent** available.

2.5.93.1 Properties

Siteld

This property contains the node name.

Siteld	
<i>String</i>	

SiteType

This property contains the node type.

SiteType	
<i>SiteTypeEnum</i>	

State

This property contains the node state.

State	
<i>SiteStateEnum</i>	

TypeDetail

This property contains detailed information on the node type.

TypeDetail	
<i>NodeTypeEnum</i>	

2.5.94 IGetAgentStateDataX

This interface makes the result data for the **GetAgentState** method and the **OnGetAgentStateEvent** available.

2.5.94.1 Properties

AgentId

This property contains the agent ID.

AgentId	
<i>String</i>	

AgentState

This property contains the agent routing state.

AgentState	
<i>UserAgentStatesEnum</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.94.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.95 IGetAgentStateOsvHuntgroupDataX

This interface makes the result data for the **GetAgentStateOsvHuntgroup** method and the **OnGetAgentStateOsvHuntgroupEvent** available.

2.5.95.1 Properties

AgentState

This property contains the agent routing state.

AgentState	
<i>AgentStateOsvHuntgroupEnum</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.95.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.96 IGetDoNotDisturbDataX

This interface makes the result data for the **GetDoNotDisturb** method and the **OnGetDoNotDisturbEvent** available.

2.5.96.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

DoNotDisturbState

This property contains the do-not-disturb state.

DoNotDisturbState	
<i>DoNotDisturbFlagEnum</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.96.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.97 IGetExtensionCallListDataX

This interface makes the result data for the **GetExtensionCallList** method and the **OnGetExtensionCallListEvent** available.

2.5.97.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

ExtensionCallList

This property contains the list of connected calls.

ExtensionCallList	
<i>IExtensionCallList</i>	

2.5.98 IGetForwardingDataX

This interface makes the result data for the **GetDoNotDisturb** method and the **OnGetDoNotDisturbEvent** available.

2.5.98.1 Properties

BusyDestination

The property contains the forwarding destination if busy.

BusyDestination	
<i>String</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Destination

The property contains the forwarding destination for immediate forwarding.

Destination	
<i>String</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

ForwardingState

This property contains the forwarding state.

ForwardingState	
<i>ForwardingStateEnum</i>	

NoAnswerDestination

This property contains the forwarding destination for time-controlled forwarding (Ring-No-Answer forwarding).

NoAnswerDestination	
String	

2.5.98.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
Return		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
Return		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
Return		
String		

2.5.99 IGetPhoneStateDataX

This interface makes the result data for the **GetPhoneState** available.

2.5.99.1 Properties

Errorcode

This property contains the error code.

Errorcode	
ErrorEnum	

PhoneStateInfoData

This property contains the telephone state data.

PhoneStateInfoData	
IPhoneStateChangeInfoDataX	

2.5.100 IHoldCallDataX

This interface makes the result data for the **HoldCall** method and the **OnHoldCallEvent** available.

2.5.100.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.100.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.101 ILoggingConfigDataX

This interface makes the general setting data for the process log available.

2.5.101.1 Properties

LogAdvancedDirectory

This property contains the path to the file directory in which the log files are saved.

LogAdvancedDirectory	
<i>String</i>	

LogAdvancedEnabled

This property indicates whether the messages sent in OnLoggingEvent are written to a SmartInspect log file.

LogAdvancedEnabled	
<i>Boolean</i>	

LogAdvancedFilename

This property contains the path of the current log file (the file name is generated automatically).

LogAdvancedFilename	
<i>String</i>	

LogAdvancedFileParameter

This property contains the control values for the SmartInspect log files.

LogAdvancedFileParameter	
<i>String</i>	

LogConnectivityMessages

This property indicates whether the messages to and from the Integration Connector should be logged in OnLoggingEvent.

LogConnectivityMessages	
<i>Boolean</i>	

LogInternalMessages

This property indicates whether internal function calls should be logged in OnLoggingEvent.

LogInternalMessages	
<i>Boolean</i>	

LogKeepAlive

This property indicates whether keep-alive messages should be logged in OnLoggingEvent.

LogKeepAlive	
<i>Boolean</i>	

LogMessageDelimiter

This property specifies the delimiter used between individual values in ToString outputs of event files.

LogMessageDelimiter	
<i>LogMessageDelimiterEnum</i>	

LogMsgDelimiterValue

This property specifies the delimiter used between individual values in ToString outputs of event files.

LogMsgDelimiterValue	
<i>String</i>	

LogTcpIpBuffer

This property indicates whether the messages exchanged via TCP/IP should be logged in OnLoggingEvent.

LogTcpIpBuffer	
<i>Boolean</i>	

LogTimerActivity

This property indicates,.

LogTimerActivity	
<i>Boolean</i>	

LogWithTimestamp

This property indicates whether the date and time value of the message should be contained in OnLoggingEvent.

LogWithTimestamp	
<i>Boolean</i>	

2.5.102 IMakeCallDataX

This interface makes the result data for the **MakeCall** method and the **OnMakeCallEvent** available.

2.5.102.1 Properties

CallId

This property contains the call ID of the call.

CallId	
<i>String</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.102.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		

2.5.103 IOldCallIdListX

This interface makes a list of call IDs available.

2.5.103.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.103.2 Methods

get_Item

This method returns the call ID.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
String		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.104 IPartyDataX

This interface makes detailed information on a call party available.

2.5.104.1 Properties

Device

This property contains the extension or the trunk number (for external subscribers).

Device	
<i>String</i>	

DeviceDisplay

This property contains the display of the extension.

DeviceDisplay	
<i>String</i>	

DeviceType

This property contains the extension type.

DeviceType	
<i>DeviceTypeEnum</i>	

Position

This property indicates the position of the extension in the list of conference participants.

Position	
<i>int</i>	

2.5.105 IPartyListX

This interface makes a list of IPartyDataX objects available.

2.5.105.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.105.2 Methods

get_Item

This methods returns the IPartyDataX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
IPartyDataX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.106 IPhoneCallStateChangeInfoDataX

This interface makes the result data for the **OnPhoneCallStateChangeEvent** available.

2.5.106.1 Properties

Action

This property specifies the action that led to the change.

Action	
<i>CallListChangeNotificationEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

ExtensionCallList

This property contains the list of connected calls.

ExtensionCallList	
<i>IExtensionCallList</i>	

Siteld

This property contains the node name.

Siteld	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

2.5.107 IPhoneClearedInfoDataX

This interface makes the information data for the **OnPhoneClearedInfoEvent** available.

2.5.107.1 Properties

CallId

This property contains the call ID of the disconnected call.

CallId	
String	

DisconnectedDevice

This property contains the phone number that was disconnected from the connection.

DisconnectedDevice	
String	

DisconnectingDevice

This property contains the phone number that disconnected the connection.

DisconnectingDevice	
String	

EventCause

This property contains the event reason for a CSTA command.

EventCause	
EventCauseEnum	

Extension

This property contains the extension.

Extension	
String	

2.5.108 IPhoneDialingInfoDataX

This interface makes the information data for the **OnPhoneDialingInfoEvent** available.

2.5.108.1 Properties

CalledDevice

This property contains the called phone number.

CalledDevice	
String	

CallId

This property contains the call ID.

CallId	
String	

Extension

This property contains the extension.

Extension	
String	

2.5.109 IPhoneRingInfoDataX

This interface makes the information data for the **OnPhoneRingInfoEvent** available.

2.5.109.1 Properties

CallId

This property contains the call ID of the disconnected call.

CallId	
String	

CallingDevice

This property contains the incoming phone number.

CallingDevice	
String	

CallWaiting

This property indicates the wait state.

CallWaiting	
Boolean	

EventCause

This property contains the event reason for a CSTA command.

EventCause	
<i>EventCauseEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

LastRedirectingDevice

This property contains the extension of the last call redirection.

LastRedirectingDevice	
<i>String</i>	

OriginalCalledDevice

This property contains the originally called number.

OriginalCalledDevice	
<i>String</i>	

RingDevice

This property contains the extension at which the call arrives.

RingDevice	
<i>String</i>	

Siteld

This property contains the node name.

Siteld	
<i>String</i>	

2.5.110 IPhoneStateBusyInfoDataX

This interface makes the information data for the **OnPhoneStateBusyInfoEvent** available.

2.5.110.1 Properties

AlertingCallId

This property contains the call ID of the alerting call.

AlertingCallId	
<i>String</i>	

BusyState

This property contains the busy state.

BusyState	
<i>BusyStateEnum</i>	

BusyStateDetail

This property contains details on the busy state.

BusyStateDetail	
<i>BusyStateDetailEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

PartyDevice

This property contains the subscriber phone number.

PartyDevice	
<i>String</i>	

Siteld

This property contains the node name.

Siteld	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

2.5.111 IPhoneStateChangeInfoDataX

This interface makes the information data for the **OnPhoneStateChangeInfoEvent** available.

2.5.111.1 Properties

CallState

This property contains the connection state.

CallState	
<i>CallStateEnum</i>	

Direction

This property contains the connection direction.

Direction	
<i>CallDirectionEnum</i>	

Display

This property contains the text of the device display.

Display	
<i>String</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

RemotePartyBusy

This property indicates whether the remote party is busy.

RemotePartyBusy	
<i>Boolean</i>	

ReplacedCallId

This property contains the call ID replaced by the call ID of the conference.

ReplacedCallId	
<i>String</i>	

ServicePermittedFlags

This property contains the current CTI functions.

ServicePermittedFlags	
<i>IServPermFlagsX</i>	

UserData

This property contains the additional user data.

UserData	
String	

2.5.112 IReconnectCallDataX

This interface makes the result data for the **ReconnectCall** method and the **OnReconnectCallEvent** available.

2.5.112.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
ICstaErrorX	

Errorcode

This property contains the error code.

Errorcode	
ErrorEnum	

Extension

This property contains the extension.

Extension	
String	

2.5.112.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
Return		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
Return		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.113 IRetrieveCallDataX

This interface makes the result data for the **RetrieveCall** method and the **OnRetrieveCallEvent** available.

2.5.113.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.113.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.114 ISendDTMFDataX

This interface makes the result data for the **SendDTMF** method and the **OnSendDTMFEvent** available.

2.5.114.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.114.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.115 IServerConnectionStateChangeDataX

This interface makes the information data for the **OnServerConnectionStateChangeEvent** available.

2.5.115.1 Properties

NewConnectionState

This property contains the current connection state.

NewConnectionState	
<i>ServerConnectionStateEnum</i>	

OldConnectionState

This property contains the previous connection state.

OldConnectionState	
<i>ServerConnectionStateEnum</i>	

ServerType

This property contains the server type.

ServerType	
<i>ServerTypeEnum</i>	

2.5.116 ISetAgentStateOsvHuntgroupDataX

This interface makes the result data for the **SetAgentStateOsvHuntgroup** method and the **OnSetAgentStateOsvHuntgroupEvent** available.

2.5.116.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.116.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.116.3 ISetForwardingDataX

This interface makes the result data for the **SetForwarding** method and the **OnSetForwardingEvent** available.

2.5.116.4 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.116.5 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.117 ISingleStepTransferCallDataX

This interface makes the result data for the **SingleStepTransferCall** method and the **OnSingleStepTransferCallEvent** available.

2.5.117.1 Properties

CallId

This property contains the call ID of the transfer call.

CallId	
<i>String</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.117.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.118 ISiteStateInfoDataX

This interface makes the result data for the **OnSiteStateInfoEvent** available.

2.5.118.1 Properties

FailoverFlag

This property indicates whether, in the case of geographically separated OSV instances, an error led to a node change.

FailoverFlag	
<i>Boolean</i>	

FunctionNodeId

This property contains the ID of the function node.

FunctionNodeId	
<i>Integer</i>	

FunctionNodeSoftwareVersion

This property contains the version number of the function node.

FunctionNodeSoftwareVersion	
<i>String</i>	

OSCCSiteKey

This property contains the key of the OSCC instance (only with the HPPC connector type).

OSCCSiteKey	
<i>String</i>	

SiteId

This property contains the node name.

SiteId	
<i>String</i>	

SiteType

This property contains the node type.

SiteType	
<i>SiteTypeEnum</i>	

State

This property contains the state of the node.

State	
<i>SiteStateEnum</i>	

TypeDetail

This property contains detailed information on the node type.

TypeDetail	
<i>NodeTypeEnum</i>	

2.5.119 ISnapshotAdvCallListX

This interface makes a list of ISnapshotAdvCallX objects available.

2.5.119.1 Properties

Count

This property contains the number of lists.

Count	
<i>Integer</i>	

2.5.119.2 Methods

get_Item

This methods returns the ISnapshotAdvCallX object.

get_Item		
<i>Parameters</i>		
Index	<i>Integer</i>	List index
<i>Return</i>		
ISnapshotAdvCallX		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

2.5.120 ISnapshotAdvCallX

This interface makes the state of a call determined by a snapshot available.

2.5.120.1 Properties

CallId

This property contains the call ID of the call.

CallId	
<i>String</i>	

Lcs

This property contains the connection state.

Lcs	
<i>LocalConnectionStateEnum</i>	

ServicePermittedFlags

This property contains the CTI functions currently available.

ServicePermittedFlags	
<i>IServPermFlagsX</i>	

2.5.120.2 Methods

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.121 ISnapshotDeviceAdvancedDataX

This interface makes the result data for the **SnapshotDeviceAdvanced** method and the **OnSnapshotDeviceAdvancedEvent** available.

2.5.121.1 Properties

CallList

This property contains the list of connected calls.

CallList	
<i>ISnapshotAdvCallListX</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
String	

2.5.121.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
Return		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
Return		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
Return		
String		

2.5.122 ISnapshotDeviceDataX

This interface makes the result data for the **SnapshotDevice** method and the **OnSnapshotDeviceEvent** available.

2.5.122.1 Properties

CallId

This property contains the call ID of the current call.

CallId	
String	

CstaError

This property contains details on the CSTA error.

CstaError	
ICstaErrorX	

CstaVersion

This property contains the CSTA log version.

CstaVersion	
<i>CstaVersionEnum</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

Lcs

This property contains the connection state.

Lcs	
<i>LocalConnectionStateEnum</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

2.5.123 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.124 IStartMonitorDeviceDataX

This interface makes the result data for the **StartMonitorDevice** method and the **OnStartMonitorDeviceEvent** available.

2.5.124.1 Properties

Busy

This property indicates whether the extension is busy.

Busy	
<i>Boolean</i>	

CallId

This property contains the call ID of the current call.

CallId	
<i>String</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

CstaVersion

This property contains the CSTA log version.

CstaVersion	
<i>CstaVersionEnum</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	

Extension	
<i>String</i>	

Lcs

This property contains the connection state.

Lcs	
<i>LocalConnectionStateEnum</i>	

MonitorDeviceHandlingFlag

This property contains details on the type of monitoring of the extension.

MonitorDeviceHandlingFlag	
<i>MonitorDeviceHandlingFlagEnum</i>	

NodeType

This property contains detailed information on the node type.

NodeType	
<i>NodeTypeEnum</i>	

Siteld

This property contains the node name.

Siteld	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

2.5.124.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.125 IStopMonitorDeviceDataX

This interface makes the result data for the **StopMonitorDevice** method and the **OnStopMonitorDeviceEvent** available.

2.5.125.1 Properties

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

CstaVersion

This property contains the CSTA log version.

CstaVersion	
<i>CstaVersionEnum</i>	

Errorcode

This property contains the error code.

Errorcode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

MonitorDeviceHandlingFlag

This property contains details on the type of monitoring of the extension.

MonitorDeviceHandlingFlag	
<i>MonitorDeviceHandlingFlagEnum</i>	

Siteld

This property contains the node name.

Siteld	
<i>String</i>	

UserData

This property contains the additional user data.

UserData	
<i>String</i>	

2.5.125.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.126 ITransferCallDataX

This interface makes the result data for the **TransferCall** method and the **OnTransferCallEvent** available.

2.5.126.1 Properties

CallId

This property contains the call ID of the transfer call.

CallId	
<i>String</i>	

CstaError

This property contains details on the CSTA error.

CstaError	
<i>ICstaErrorX</i>	

ErrorCode

This property contains the error code.

ErrorCode	
<i>ErrorEnum</i>	

Extension

This property contains the extension.

Extension	
<i>String</i>	

2.5.126.2 Methods

IsCstaError

This method indicates whether a CSTA error occurred.

IsCstaError		
<i>Return</i>		
Boolean		

IsError

This method indicates whether an error occurred.

IsError		
<i>Return</i>		
Boolean		

ToString

This method returns a string that indicates the current object.

ToString		
<i>Return</i>		
String		

2.5.127 ITransferCompletedInfoDataX

This interface makes the result data for the **OnTransferCompletedInfoEvent** available.

2.5.127.1 Properties

CallId

This property contains the call ID of the transfer call.

CallId	
<i>String</i>	

ConnectionList

This property contains the list of subscribers connected after the transfer.

ConnectionList	
<i>IConnectionListX</i>	

EventCause

This property contains the event reason of the CSTA command.

EventCause	
<i>EventCauseEnum</i>	

Extension

This property contains the transferred extension.

Extension	
<i>String</i>	

LastRedirectionDevice

This property contains the extension of the last call redirection.

LastRedirectionDevice	
<i>String</i>	

Lcs

This property contains the connection state.

Lcs	
<i>LocalConnectionStateEnum</i>	

OldCallIdOnHold

This property contains the call ID of the original call.

OldCallIdOnHold	
<i>String</i>	

OldCallIdTransferTo

This property contains the call ID of the consultation call.

OldCallIdTransferTo	
<i>String</i>	

Siteld

This property contains the node name.

Siteld	
<i>String</i>	

TransferredDevice

This property contains the transferred extension.

TransferredDevice	
<i>String</i>	

TransferredToDevice

This property contains the called number of the transfer.

TransferredToDevice	
<i>String</i>	

TransferringDevice

This property contains the transferring extension.

TransferringDevice	
<i>String</i>	

2.5.128 ITransferredInfoDataX

This interface makes the result data for the **OnTransferredInfoEvent** available.

2.5.128.1 Properties

CallId

This property contains the call ID of the transfer call.

CallId	
<i>String</i>	

ConnectionList

This property contains the list of subscribers connected after the transfer.

ConnectionList	
<i>IConnectionListX</i>	

EventCause

This property contains the event reason of the CSTA command.

EventCause	
<i>EventCauseEnum</i>	

Extension

This property contains the transferred extension.

Extension	
<i>String</i>	

LastRedirectionDevice

This property contains the extension of the last call redirection.

LastRedirectionDevice	
<i>String</i>	

Lcs

This property contains the connection state.

Lcs	
<i>LocalConnectionStateEnum</i>	

OldCallId

This property contains the call ID of the consultation call.

OldCallId	
String	

SiteId

This property contains the node name.

SiteId	
String	

TransferredDevice

This property contains the transferred extension.

TransferredDevice	
String	

TransferredToDevice

This property contains the called number of the transfer.

TransferredToDevice	
String	

TransferringDevice

This property contains the transferring extension.

TransferringDevice	
String	

2.5.129 IUserAgentServPermFlagsX

This interface makes the possible agent state functions available.

2.5.129.1 Properties

Available

This property indicates whether the agent could change to the "available" state.

Available	
Boolean	

Logon

This property indicates whether the agent could change to the "logged on" state.

Logon	
<i>Boolean</i>	

Logoff

This property indicates whether the agent could change to the "logged off" state.

Logoff	
<i>Boolean</i>	

Unavailable

This property indicates whether the agent could change to the "unavailable" state.

Unavailable	
<i>Boolean</i>	

Work

This property indicates whether the agent could change to the "work" state.

Work	
<i>Boolean</i>	

3 Realtime

3.1 Basics

The real-time interface enables access to the real-time data of a contact center. This interface can be used to collect real-time data of the contact center.

There are three types of real-time information.

- **Queues**
These real-time data deliver the current state of a queue. This includes how many agents are currently available in the status, information on the accessibility of the queue and the average wait period for incoming calls.
- **Groups**
These data contain the current figures on the handled tasks within a group and information on the current load.

NOTE:

Groups are only used in connection with Genesys.

- **Agents**
In addition to general information, these data contain, for example, the average processing time of a task and the current state of an agent.

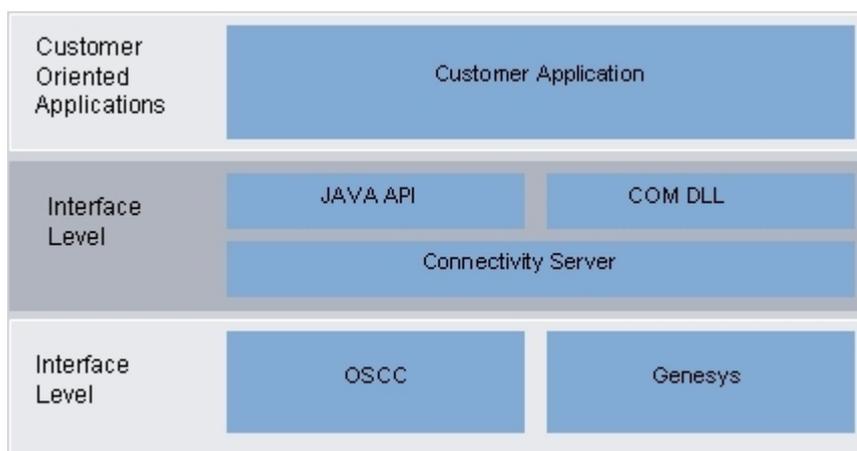
3.1.1 System architecture

The data described above can be used via the interface library.

The interface is available as a COM DLL and Java API.

The interface library can therefore be used in any programming language that exists for a COM implementation, e.g. Visual Basic®, Visual C#® and Delphi®. The Java API is used in Java projects.

The following figure shows how the information flow between the various layers in the architecture is structured.



The **Platform Level** contains the basic physical system. Two different contact centers are shown here, namely the OSCC (OpenScape ContactCenter) and Genesys.

The **Interface Level** provides the interfaces for the customer, ultimately making it possible to use any system on the "Platform Level".

The implementation of the various systems is handled by the **Connectivity Server**. The Connectivity Server with the interfaces is the middleware in this layer model. With the aid of the Connectivity Server, the interfaces deliver data that have already been processed and can be simply integrated in an application.

This three-layer architecture simplifies access to the real-time data of a contact center and also makes it possible to use your own solution with various contact centers.

3.1.2 Application example COM DLL

The following examples are written in C#. Usage is similar in other COM-compatible programming languages.

3.1.2.1 Registration of the interface library

Before the interface library can be used, it must be registered in the operating system. This can take place in the development environment or directly under Windows with the following command, for example:

```
regsvr32.exe cocrealtimecust.dll
```

During development, the COM DLL must be imported into the project.

3.1.2.2 Establishing a connection with the Connectivity Server

To establish a connection with the Connectivity Server, an object of the *RealtimeLink* class is required. Before the *OpenConnection()* function can be called, the connection data must be passed to the object.

1	<code>this.realtimeLink = new RealtimeLink();</code>
2	<code>this.realtimeLink.Address = ip;</code>
3	<code>this.realtimeLink.Port = port;</code>

In this example, a *realtimeLink* class variable is generated and the parameters *Address* (IP address) and *Port* (port address) are passed. After instantiation of the object, the connection to the Connectivity Server can be established. Before the connection is established, it is advisable to implement the events of the *RealtimeLinks* to be able to respond to events of the interface later on.

1	<code>this.realtimeLink.OnAgentConfigReceived += new IRealtimeLinkEvents_OnAgentConfigReceivedEventHandler (realtimeLink_OnAgentConfigReceived);</code>
2	<code>this.realtimeLink.OnAgentDataReceived += new IRealtimeLinkEvents_OnAgentDataReceivedEventHandler (realtimeLink_OnAgentDataReceived);</code>
3	<code>this.realtimeLink.OnError += new IRealtimeLinkEvents_OnErrorEventHandler (realtimeLink_OnError);</code>
4	<code>this.realtimeLink.OnGroupConfigReceived += new IRealtimeLinkEvents_OnGroupConfigReceivedEventHandler (realtimeLink_OnGroupConfigReceived);</code>
5	<code>this.realtimeLink.OnGroupDataReceived += new IRealtimeLinkEvents_OnGroupDataReceivedEventHandler (realtimeLink_OnGroupDataReceived);</code>

```

1  this.realtimeLink.OnAgentConfigReceived += new
    IRealtimeLinkEvents_OnAgentConfigReceivedEventHandler
    (realtimeLink_OnAgentConfigReceived);
6  this.realtimeLink.OnLogoffResponse += new
    IRealtimeLinkEvents_OnLogoffResponseEventHandler
    (realtimeLink_OnLogoffResponse);
7  this.realtimeLink.OnLogonResponse += new
    IRealtimeLinkEvents_OnLogonResponseEventHandler
    (realtimeLink_OnLogonResponse);
8  this.realtimeLink.OnQueueConfigReceived += new
    IRealtimeLinkEvents_OnQueueConfigReceivedEventHandler
    (realtimeLink_OnQueueConfigReceived);
9  this.realtimeLink.OnQueueDataReceived += new
    IRealtimeLinkEvents_OnQueueDataReceivedEventHandler
    (realtimeLink_OnQueueDataReceived);
10 this.realtimeLink.OnServerStateChange += new
    IRealtimeLinkEvents_OnServerStateChangeEventHandler
    (realtimeLink_OnServerStateChange);
11 this.realtimeLink.OnTeamConfigReceived += new
    IRealtimeLinkEvents_OnTeamConfigReceivedEventHandler
    (realtimeLink_OnTeamConfigReceived);
12 this.realtimeLink.OnDebug += new
    IRealtimeLinkEvents_OnDebugEventHandler
    (realtimeLink_OnDebug);

```

In this example, all events of the *RealtimeLinks* are evaluated. For this purpose, one event handler was added to each event.

In the next step, a connection is established to the Connectivity Server.

```

1  EnumErrorCode errorCode = this.realtimeLink.OpenConnection();

```

The *OpenConnection()* function returns an enumeration of type *EnumErrorCode* and indicates, if applicable, whether the connection was established. If the connection was established, *NO_ERROR* is returned.

Because each user can be assigned different access authorizations, a login must be performed. As soon as the *realtimeLink_OnServerStateChanged* event is triggered and the delivered state is *SST_UP*, a login can take place.

```

1  void realtimeLink_OnServerStateChange(EnumServerState State)
2  {
3      if (State.Equals(EnumServerState.SST_UP))
4      {
5          ILogonResponse logonResponse =
            this.realtimeLink.Logon(AgentId, AgentPassword, false);
7      }
8  }

```

The login is performed via the *Logon* method. The Agent ID, Agent Password and a Boolean for synchronous execution of the method are passed as parameters. The method returns a *ILogonResponse* that delivers detailed information on the login process.

3.1.2.3 Receiving data

In a fixed time interval, the Connectivity Server sends the current state of the various data sources after the login. The data of the queue can be received as follows.

Queues

Receiving and processing the information of the queues:

```
1 void realtimeLink_OnQueueDataReceived(IQueuesData QueuesData)
2 {
3     for (int i = 1; i < QueuesData.QueueObjectsList.Count; i++)
4     {
5         IQueueObject queue =
6             QueuesData.QueueObjectsList.get_Item(i);
7         string queueName = queue.QueueName;
8         string accessibility = queue.QueueAccessibility;
9         string averageWaitTime = queue.AverageWaitTime;
10    }
11 }
12 }
```

The event brings along an object of the *IQueuesData* class. This object contains the current states of the queues that the logged-in user is permitted to see. This example shows how a *for* loop can be used to access the various queues and their parameters. In this example, the queue name (*QueueName*), the accessibility (*QueueAccessibility*) and the average wait time of a task (*AverageWaitTime*) are read out.

3.1.2.4 Agents

Receiving and processing the information of the agents:

```
1 void realtimeLink_OnAgentDataReceived(IAgentsData AgentsData)
2 {
3     for (int i = 1; i <= AgentsData.AgentObjectsList.Count; i++)
4     {
5         IAgentObject agent =
6             AgentsData.AgentObjectsList.get_Item(i);
7         string id = agent.AgentId;
8         string extension = agent.ExtensionNumber;
9         EnumAgentState state = agent.AgentState;
10    }
11 }
```

The event delivers an object of the *IAgentData* class. This object contains an array with agents. As with the queues, the list is run through with a "for" loop. In this example, the agent ID (*AgentId*), the extension (*ExtensionNumber*) and the current state of the agent (*AgentState*) are read out. The various states that an agent can take on are defined in the *EnumAgentState* enumeration.

Groups

Receiving and processing the information of the groups:

```

1 void realtimeLink_OnGroupDataReceived(IGroupsData GroupsData)
2 {
3     for (int i = 1; i < GroupsData.GroupObjectsList.Count; i++)
4     {
5         IGroupObject group =
6             GroupsData.GroupObjectsList.get_Item(i);
7         string groupName = group.GroupName;
8         string available = group.TotalAgentsAvailable;
9         string unavailable = group.TotalAgentsIdle;
10        string wrapup = group.TotalAgentsWrapup;
11    }
12 }

```

The processing of group information is similar to the evaluations used for queues and agents. In this example, the group name (*GroupName*), the number of agents that are available (*TotalAgentsAvailable*), the number of agents that are unavailable (*TotalAgentsIdle*) and the number of agents that are currently in the wrapup state (*TotalAgentsWrapup*) are read out.

3.1.3 Application development with Java API

The following examples are written in Java.

3.1.3.1 Integration of the Java API

Before using the Java API, it must be integrated in the project. This can take the form of a JAR or external JAR archive.

3.1.3.2 Establishing a connection with the Connectivity Server

To establish a connection with the Connectivity Server, an object of the *RealtimeConfigData* class is required. The connection data are passed to this object, and the object later serves as the parameter for an object of the *RealtimeClient* class.

At least the following connection parameters must be specified.

```

1 RealtimeConfigData realtimeConfigData = new
  RealtimeConfigData();
2 realtimeConfigData.setIpAddressRealtime( xxx.xxx.xxx.xxx );
3 realtimeConfigData.setIpPortRealtime(7498);

```

After instantiation of the object, an object of the *RealtimeClient* class can be created and the connection to the Connectivity Server can be established. Before establishing the connection, it is advisable to implement the *RealtimeListener* to be able to respond to events of the API.

```

1 RealtimeClient mRealtime = new
  RealtimeClient(realtimeConfigData);
2 mRealtime.addRealtimeListener(this);
3 ErrorCodeEnum errorCodeEnum = mRealtime.openConnection();

```

The *openConnection()* function returns an enumeration of type *ErrorCodeEnum* and indicates, if applicable, whether the connection was established. If the connection was established, *NO_ERROR* is returned.

Because each user can be assigned different access authorizations, a login must be performed. As soon as the *handleServerStateEvent* event is triggered and the delivered state is *SST_UP*, a login can take place.

```
1 public void handleServerStateEvent(ServerStateEvent event)
2 {
3     if(event.getState() == RealtimeServerStateEnum.SST_UP){
4         LogonResponse logOnResponse = mRealtime.logon(AgentId,
5             AgentPassword, true);
6     }
7 }
```

The login is performed via the *logon* method. The Agent ID, Agent Password and a Boolean for synchronous execution of the method are passed as parameters. The method returns a *LogonResponse* that delivers detailed information on the login process.

3.1.3.3 Receiving data

In a fixed time interval, the Connectivity Server sends the current state of the various data sources after the login. The data of the queue can be received as follows.

Queues

Receiving and processing the information of the queues:

```
1 public void handleQueueDataReceived(QueueDataEvent event)
2 {
3     QueueDataEvent evt = event;
4     for (int i = 0; i < evt.getQueues().size(); i++)
5     {
6         Queue queue = evt.getQueues().get(i);
7         String queueName = queue.getQueueName();
8         String accessibility = queue.getAccessibility();
9         String averageWaitTime = queue.getAverageWaitTime();
10    }
11 }
```

The event brings along an object of the *QueueDataEvent* class. This object contains the current states of the queues that the logged-in user is permitted to see. This example shows how a "for" loop can be used to access the various queues and their parameters. In this example, the queue name (*getQueueName*), the accessibility (*getAccessibility*) and the average wait time of a task (*getAverageWaitTime*) are read out.

Agents

Receiving and processing the information of the agents:

```
1 public void handleQueueDataReceived(QueueDataEvent event)
2 {
3     AgentDataEvent evt = event;
4     for (int i = 0; i < evt.getAgents().size(); i++)
5     {
6         Agent agent = evt.getAgents().get(i);
7         String id = agent.getAgentId();
8         String name = agent.getExtension();
9         AgentStateEnum state = agent.getAgentState();
10    }
11 }
```

The event delivers an object of the *AgentDataEvent* class. This object contains an array with agents. As with the queues, the list is run through with a "for" loop. In this example, the agent ID (*getAgentId*), the extension (*getExtension*) and the current state of the agent (*getAgentState*) are read out. The various states that an agent can take on are defined in the *AgentStateEnum* enumeration.

Groups

Receiving and processing the information of the groups:

```
1 public void handleGroupDataReceived(GroupDataEvent event)
2 {
3     GroupDataEvent evt = event;
4     for (int i = 0; i < evt.getGroups().size(); i++)
5     {
6         Group group = evt.getGroups().get(i);
7         String groupName = group.getGroupName();
8         String available = group.getTotalAgentsAvailable();
9         String unavailable = group.getTotalAgentsIdle();
10        String wrapup = group.getTotalAgentsWrapup();
11    }
12 }
```

The processing of group information is similar to the evaluations used for queues and agents. In this example, the group name (*getGroupName*), the number of agents that are available (*getTotalAgentsAvailable*), the number of agents that are unavailable (*getTotalAgentsIdle*) and the number of agents that are currently in the wrapup state (*getTotalAgentsWrapup*) are read out.

3.2 Enumeration types

3.2.1 EnumAgentColumnType

The enumeration type describes all possible agent columns.

Table: EnumAgentColumnType

EnumAgentColumnType	
AGENT_ID	Agent ID.
AGENT_IDENTIFIER	Agent's unique identifier, consists of OSCC ID and Agent ID.
AGENT_STATE	Agent's status
AGENT_TIME_IN_STATE	Agent's time in state.
AVERAGE_TASK_TIME	Agent's average task time.
CALL_TYPE	Agent's call type
CUSTOM_VALUE1	
CUSTOM_VALUE10	
CUSTOM_VALUE2	
CUSTOM_VALUE3	
CUSTOM_VALUE4	
CUSTOM_VALUE5	
CUSTOM_VALUE6	
CUSTOM_VALUE7	
CUSTOM_VALUE8	
CUSTOM_VALUE9	
EXTENSION	Agent's extension.
FIRST_NAME	Agent' first name.
HANDLED_TASKS	Agent's handled tasks.
LAST_NAME	Agent's last name
REASON_CODE	Agent's reason code.
RESERVED1	
RESERVED10	
RESERVED2	
RESERVED3	
RESERVED4	
RESERVED5	
RESERVED6	
RESERVED7	
RESERVED8	
RESERVED9	
TEAM_KEY	Agent's team key

3.2.2 EnumAgentState (only OSCC V7)

The enumeration type describes all possible agent states.

Table: EnumAgentState

EnumAgentState	
AVAILABLE	Agent is logged on and available for calls.
BUSY	Agent's extension is busy.
CONSULTING	Agent is consulting.
DIALING	Agent is dialing an outgoing call.
HOLDING	Agent's extension is in hold state.
LOGGED_OFF	Agent is logged off.
LOGGED_ON	Agent is logged on.
OUT_OF_SERVICE	Agent's extension is out of service (e.g.: extension could be unplugged, the phone may be defective, etc).
PENDING	Agent has a call and is pending for them.
PROCESSING	Agent is processing the current call.
QUEUED	Agent is queued for a conference.
RINGING	Agent's extension is ringing.
TALKING	Agent is talking.
UNKOWN	Unknown agent state.
WORKING	Agent is logged on and in work state.

3.2.3 EnumCallType

This enumeration type describes all call types.

Table: EnumCallType

EnumCallType	
CONTACT_TYPE_MAX	Contact type max.
DIRECT_INCOMING_VOICE	Direct incoming voice call.
DIRECT_INTERNAL_VOICE	Direct internal voice call.
DIRECT_OUTGOING_EMAIL	Direct outgoing email.
DIRECT_OUTGOING_VOICE	Direct outgoing voice call.
NONE	No call.
ROUTED_CALLBACK	Routed call back call.
ROUTED_EMAIL	Routed email.
ROUTED_VOICE	Routed voice call.
ROUTED_WEB_COLLABORATION	Routed web collaboration.

3.2.4 EnumDataType

The enumeration type describes all possible data types.

Table: EnumDataType

EnumDataType	
DOUBLE_TYPE	Double data type.
INT_TYPE	Integer data type.
PERCENT_TYPE	Percentage data type.
STRING_TYPE	Direct outgoing email.
TIME_TYPE	Time data type.

3.2.5 EnumErrorCode

The enumeration type describes all possible error codes.

Table: EnumErrorCode

EnumErrorCode	
CANNOT_CHECK_PASSWORD	Unable to check password.
CONN_ALREADY_OPEN	There is already an open connection.
DATABASE_ERROR	Database error.
GLOBAL	Global error.
HEARTBEAT_NO_ANSWER	No answer to heartbeat.
INTERNAL_UNKNWON	Internal unknown error.
KDO_TIMEOUT	Commando timeout.
LINK_NOT_UP	Link is down.
LOGGED_IN_OTHER_ADDR	Already logged in with same login on another ip address.
LOGGED_IN_SAME_ADDR	Already logged in with same login on same ip address.
LOGIN_FAILED	Login failed.
LOGIN_UNKNOWN	Login id unknown.
MSG_TYPE_INVALID	Message type is invalid.
NO_CONNECTION_TO_DB	There is no connection to database.
NO_DATA_AVAILABLE	There is no data available.
NO_ERROR	Everything seems to be fine.
PROP_NOT_VALID	Property is not valid.
REGISTER_FAILED	Register failed.
SOCKET	Socket error.
UNKNWON_MSG_TYPE	Message type is unknown.
WRONG_PASSWORD	Password seems to be wrong.

3.2.6 EnumGroupColumnType

The enumeration type describes all possible group columns.

Table: EnumGroupColumnType

EnumGroupColumnType	
CUSTOM_VALUE1	
CUSTOM_VALUE10	
CUSTOM_VALUE2	
CUSTOM_VALUE3	
CUSTOM_VALUE4	
CUSTOM_VALUE5	
CUSTOM_VALUE6	
CUSTOM_VALUE7	
CUSTOM_VALUE8	
CUSTOM_VALUE9	
NOT_USED_10	Group name.
NOT_USED_11	Agent's first name.
NOT_USED_12	Agent's handled tasks.
NOT_USED_13	Agent's last name
NOT_USED_2	Agent's reason code.
NOT_USED_3	
NOT_USED_4	
NOT_USED_5	
NOT_USED_6	
NOT_USED_7	
RESERVED1	
RESERVED10	
RESERVED2	
RESERVED3	
RESERVED4	
RESERVED5	
RESERVED6	
RESERVED7	
RESERVED8	
RESERVED9	
TOTAL_AGENTS_AVAILABLE	Group total agents available.
TOTAL_AGENTS_IDLE	Group total agents unavailable.
TOTAL_AGENTS_WRAPUP	Group total agents wrap up.
TOTAL_HANDLED	Group total handled tasks.

3.2.7 EnumLoggedOnMedia

This enumeration type describes all possible media for which an agent can be logged on. *Note: effective as of V5.0.0.0.*

Table: EnumLoggedOnMedia

EnumLoggedOnMedia	
UNKNOWN	unknown
VOICE	voice
CALLBACK	Callback
VOICE_CALLBACK	Voice and Callback
EMAIL	Email
VOICE_EMAIL	Voice and Email
CALLBACK_EMAIL	Callback and Email
VOICE_CALLBACK_EMAIL	Voice and Callback and Email
WEBCHAT	Web Chat
VOICE_WEBCHAT	Voice and WebChat
CALLBACK_WEBCHAT	Callback and WebChat
VOICE_CALLBACK_WEBCHAT	Voice and Callback and WebChat
EMAIL_WEBCHAT	Email and WebChat
VOICE_EMAIL_WEBCHAT	Voice and Email and WebChat
CALLBACK_EMAIL_WEBCHAT	Callback and Email and WebChat
VOICE_CALLBACK_EMAIL_WEBCHAT	Voice and Callback and Email and WebChat

3.2.8 EnumMediaType

The enumeration type describes all possible media types.

Table: EnumLoggedOnMedia

EnumMediaType	
ALL	All the MediaTypes.
CALLBACK	A callback call
EMAIL	An email call.
VOICE	An inbound voice call.
WEBCHAT	A chat session.

3.2.9 EnumNodeType

The enumeration type describes all possible node types.

Table: EnumNodeType

EnumNodeType	
GENESYS	Genesys node
HPPC	HPPC / OSCC Node.

3.2.10 EnumQueueColumnType

The enumeration type describes all possible queue columns.

Table: EnumQueueColumnType

EnumQueueColumnType	
ACCESSIBILITY	Queue's accessibility
AVERAGE_WAIT_TIME	Queue's average wait time.
CUSTOM_VALUE1	
CUSTOM_VALUE10	
CUSTOM_VALUE2	
CUSTOM_VALUE3	
CUSTOM_VALUE4	
CUSTOM_VALUE5	
CUSTOM_VALUE6	
CUSTOM_VALUE7	
CUSTOM_VALUE8	
CUSTOM_VALUE9	
LONGEST_ITEM	Queue's longest waiting task.
QUEUE_NAME	Queue name.
RESERVED1	
RESERVED10	
RESERVED2	
RESERVED3	
RESERVED4	
RESERVED5	
RESERVED6	
RESERVED7	
RESERVED8	
RESERVED9	
SERVICE_LEVEL	Queue's service level.
TERMINATION_RATE	Queue's termination rate.
TOTAL_AGENTS_AVAILABLE	Queue's total agents available.
TOTAL_AGENTS_BUSY	Queue's total busy agents.
TOTAL_AGENTS_IDLE	Queue's total agents unavailable.
TOTAL_AGENTS_LOGGED_IN	Queue's total agents logged in.
TOTAL_AGENTS_WRAPUP	Queue's total agents wrap up.
TOTAL_DEFLECT	Queue's total deflect tasks.
TOTAL_DISCONNECTED	Queue's total disconnected tasks.
TOTAL_DISCONNECTED_IN_SERVICE	Queue's total disconnected calls within the service level interval.
TOTAL_HANDLED	Queue's total handled tasks.

EnumQueueColumnType	
TOTAL_HANDLED_IN_SERVICE	Queue's total handled calls within the service level interval.
TOTAL_OFFERED	Queue's total offered tasks.
WAITING_CONTACTS	Queue's waiting contacts.

3.2.11 EnumServerErrorType

This enumeration type describes the "User logged in a second time" error.

Table: EnumServerErrorType

EnumServerErrorType	
USER_LOGGED_IN_SECOND_TIME	User logged in at server second time

3.2.12 EnumServerState

The enumeration type describes all possible server states.

Table: EnumServerState

EnumServerState	
SST_CONNECTED	Server connected.
SST_CONNECTING	Connecting to server.
SST_DISCONNECTING	Server disconnected.
SST_DOWN	Server is down.
SST_UP	Server is up.

3.2.13 EnumUserHandlingState (only OSCC V8)

The enumeration type describes all possible agent handling states.

Table: EnumUserHandlingState

EnumUserHandlingState	
Busy	Agent's extension is busy
Consulting	Agent is consulting.
Dialing	Agent is dialing an outgoing call.
Holding	Agent's extension is in hold state.
None	None
Pending	Agent has a call and is pending for them.
Post_processing	Agent is ibusy in post-processing
Processing	Agent is processing the current call
Queued	Agent is queued for a conference.
Ringling	Agent's extension is ringing.
Talking	Agent is talking.
Unknown	Unknown agent state.

3.2.14 EnumUserPresenceState (only OSCC V8)

The enumeration type describes all possible agent presence states.

Table: EnumUserPresenceState

EnumUserPresenceState	
Away	Agent has no active contacts and his routing state is Unavailable
Busy	Agent has no active contacts and his routing state is Work
Idle	Agent has no active contacts and his routing state is Available
Notloggedon	Agent is not logged on to any media type.
Unknown	Agent is in unknown state
Active	Agent is handling one or more contacts.

3.2.15 EnumUserRoutingState (only OSCC V8)

The enumeration type describes all possible agent routing states.

Table: EnumUserRoutingState

EnumUserRoutingState	
Available	Agent is logged on to one of the contact media and is free to receive routed contacts.
Notloggedon	Agent is not logged on to any media type.
Unavailable	Agent is logged on to one of the contact media but is not available to handle routed contacts. For example, he might be having lunch or taking a break.
Unknown	Agent is in unknown state
Work	Agent is logged on to one of the contact media but is performing a work-related task and is not available to handle routed contacts. For example, he might be attending a meeting or training session.

3.2.16 EnumValid

This enumeration type describes the validity flag.

Table: EnumServerState

EnumServerState	
NOT_VALID	Real time data is not valid.
VALID	Real time data is valid.

3.3 Classes

3.3.1 IRealtimeLink

Main class of the real-time interface.

3.3.1.1 Properties

Address

IP address of the real-time server.

Address	
<i>String</i>	

Address_CB1

IP address of the first Connect Balancing server.

Address_CB1	
<i>String</i>	

Address_CB2

IP address of the second Connect Balancing server.

Address_CB2	
<i>String</i>	

AutoReconnect

Configures reconnect, true enables reconnect, false disables reconnect.

AutoReconnect	
<i>bool</i>	

DebugAllPublicMethods

This property activates logging for all public methods.

DebugAllPublicMethods	
<i>bool</i>	

DebugIgnoreHeartbeat

When this property is set to "true", heartbeat messages are ignored during logging.

DebugIgnoreHeartbeat	
<i>bool</i>	

DebugInternalMethods

This property enables logging for the internal methods of the interface.

DebugInternalMethods	
<i>bool</i>	

DebugTcp

This property enables TCP logging.

DebugTcp	
<i>bool</i>	

Port

Port number of the real-time server.

Port	
<i>Int</i>	

Port_CB1

Port number of the first Connect Balancing server.

Port CB1	
<i>Int</i>	

Port_CB2

Port number of the second Connect Balancing server.

Port CB2	
<i>Int</i>	

Region_CB

This property sets the region that may be required for Connect Balancing, depending on the configuration.

Region_CB	
<i>string</i>	

UseConnectBalancing

Enables or disables Connect Balancing.

UseConnectBalancing	
<i>bool</i>	

3.3.1.2 Methods

CloseConnection

This method closes the current connection to the OSCC-E real-time server.

CloseConnection		
<i>Return</i>		
EnumErrorCode		

Logoff

This method logs off the agent.

Logoff		
<i>Parameters</i>		
AgentId	<i>string</i>	Agent ID
Sync	<i>bool</i>	Synchronous execution
<i>Return</i>		
ILogoffResponse		

Logon

This method logs on the agent.

Logon		
<i>Parameters</i>		
AgentId	<i>string</i>	Agent ID
Password	<i>string</i>	Password
Sync	<i>bool</i>	Synchronous execution
<i>Return</i>		
ILogoffResponse		

OpenConnection

This method opens the connection to the OSCC-E real-time server.

OpenConnection		
<i>Return</i>		
EnumErrorCode		

3.3.1.3 Events

OnAgentConfigReceived

Event is output when the agent configuration from the server is received.

OnAgentConfigReceived		
<i>Parameter</i>		
ConfigData	<i>IAgentConfig</i>	Agent configuration

OnAgentDataReceived

Event is output when the agent data from the server are received.

OnAgentDataReceived		
Parameter		
AgentsData	<i>IAgentsData</i>	Agent data

OnConfiguredViewsReceived

Event is output when configured data from the server are received.

OnConfiguredViewsReceived		
Parameter		
ConfiguredViews	<i>IConfiguredViewsEvent</i>	Configured views

OnDebug

Event is output when the debug information from the server is received.

OnDebug		
Parameter		
Message	<i>string</i>	Message

OnError

Event is output when the error information from the server is received.

OnError		
Parameter		
Message	<i>string</i>	Message
Module	<i>string</i>	Module
Nr	<i>Int</i>	Number

OnGroupConfigReceived

Event is output when the group configuration from the server is received.

OnGroupConfigReceived		
Parameter		
ConfigData	<i>IGroupConfig</i>	Group configuration

OnGroupDataReceived

Event is output when the group data from the server are received.

OnGroupDataReceived		
Parameter		
GroupsData	<i>IGroupsData</i>	Group data

OnLogoffResponse

Event is output when the logoff response message is received.

OnLogoffResponse		
<i>Parameter</i>		
Response	<i>ILogoffResponse</i>	Logoff response

OnLogonResponse

Event is output when the logon response message is received.

OnLogonResponse		
<i>Parameter</i>		
Response	<i>ILogonResponse</i>	Logon response

OnQueueConfigReceived

Event is output when the queue configuration from the server is received.

OnQueueConfigReceived		
<i>Parameter</i>		
ConfigData	<i>IQueueConfig</i>	Queue configuration

OnQueueDataReceived

Event is output when the queue data from the server are received.

OnQueueDataReceived		
<i>Parameter</i>		
QueuesData	<i>IQueuesData</i>	Queue data

OnServerErrorReceived

Event is output when an error message from the server is received.

OnServerErrorReceived		
<i>Parameter</i>		
Server Error Event	<i>IServerErrorEvent</i>	Server Error

OnServerStateChange

Event is output when the general server status changes.

OnServerStateChange		
<i>Parameter</i>		
State	<i>EnumServerState</i>	Server State

OnTeamConfigReceived

Event is output when the team configuration from the server is received.

OnTeamConfigReceived		

OnTeamConfigReceived		
<i>Parameter</i>		
TeamConfig	<i>ITeamConfig</i>	Team configuration

OnVirtualAgentDataReceived

Event is output when the virtual agent message from the server is received. The VirtualAgentData object contains a list with all of the available queues.

OnVirtualAgentDataReceived		
<i>Parameter</i>		
VirtualAgentData	<i>IVirtualAgentsData</i>	Virtual Agent Data

3.4 Interfaces

3.4.1 IAcdContact

This interface makes the framework for an Acd contact available.

3.4.1.1 Properties

IsActive

This property indicates whether the contact is active:

IsActive	
<i>boolean</i>	

CallType

This property contains the call type.

CallType	
<i>EnumCallType</i>	

TimeInUserHandlingState (only OSCC V8)

This property shows the time in the current state (handling) in seconds.

TimeInUserHandlingState	
<i>Integer</i>	

UserHandlingState (only OSCC V8)

This property indicates the agent handling state.

UserHandlingState	
<i>EnumUserHandlingState</i>	

UserHandlingStateReason (only OSCC V8)

This property indicates the agent handling state reason.

UserHandlingStateReason	
<i>string</i>	

3.4.2 IAcdContactList

This interface makes the model for a list of ACD contact object available.

3.4.2.1 Properties

Count

This property contains the number of lists.

Count	
<i>int</i>	

3.4.2.2 Methods

get_Item

This methods returns the ACD contact object.

get_Item		
<i>Parameters</i>		
Index	<i>int</i>	column index
<i>Return</i>		
IAcdContact		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

3.4.3 IAcdStep

This interface makes the framework for the steps in ACD available.

3.4.3.1 Properties

TotalAgentsAvailable

This property contains the number of available agents in the queue.

TotalAgentsAvailable	
<i>Integer</i>	

TotalAgentsBusy

This property contains the number of agents with busy status in the queue.

TotalAgentsBusy	
<i>Integer</i>	

TotalAgentsIdle

This property contains the number of unavailable agents in the queue.

TotalAgentsIdle	
<i>Integer</i>	

TotalAgentsLoggedIn

This property contains the number of logged-on agents in the queue.

TotalAgentsLoggedIn	
<i>Integer</i>	

TotalAgentsPostProcessing (only OSCC V8)

This property contains the number of agents with post-processing status in the queue.

TotalAgentsPostProcessing	
<i>Integer</i>	

TotalAgentsWrapUp

This property contains the number of agents with wrapup status in the queue.

TotalAgentsWrapUp	
<i>Integer</i>	

3.4.4 IAcdStepList

This property contains a list of all AcdStep objects in the queue. The first AcdStep object contains the accumulated values of all individual steps, i.e. in a queue with two configured steps, the list contains three elements.

3.4.4.1 Properties

Count

This property contains the number of lists.

Count	
<i>int</i>	

3.4.4.2 Methods

get_Item

This method returns the Step object in the ACD.

get_Item		
<i>Parameters</i>		
Index	<i>int</i>	column index
<i>Return</i>		
IAcdStep		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

3.4.5 IAgentColumnConfig

This interface makes the framework for the agent column configuration available.

3.4.5.1 Properties

ColumnIndex

This property contains the column index.

ColumnIndex	
<i>int</i>	

ColumnType

This property contains the column type.

ColumnType	
<i>EnumAgentColumnConfig</i>	

DataType

This property contains the data type.

DataType	
<i>EnumDataType</i>	

HeaderCustom

This property contains the customized header.

HeaderCustom	
<i>string</i>	

HeaderEnglish

This property contains the English header.

HeaderEnglish	
<i>string</i>	

HeaderFrance

This property contains the French header.

HeaderFrance	
<i>string</i>	

HeaderGerman

This property contains the German header.

HeaderGerman	
<i>string</i>	

HeaderSpain

This property contains the Spanish header.

HeaderSpain	
<i>string</i>	

MaxWidth

This property contains the maximum width of the column.

MaxWidth	
<i>int</i>	

RESERVED1

This property is reserved for future implementations and is not yet in use.

RESERVED1	
<i>string</i>	

RESERVED2

This property is reserved for future implementations and is not yet in use.

RESERVED2	
<i>string</i>	

RESERVED3

This property is reserved for future implementations and is not yet in use.

RESERVED3	
<i>string</i>	

RESERVED4

This property is reserved for future implementations and is not yet in use.

RESERVED4	
<i>string</i>	

RESERVED5

This property is reserved for future implementations and is not yet in use.

RESERVED5	
<i>string</i>	

3.4.6 IAgentColumnConfigList

This interface makes the framework for a list of IAgentColumnConfig objects available.

3.4.6.1 Properties

Count

This property contains the number of lists.

Count	
<i>int</i>	

3.4.6.2 Methods

get_Item

This methods returns the agent column configuration (AgentColumnConfig) object.

get_Item		
<i>Parameters</i>		
Index	<i>int</i>	column index
<i>Return</i>		
IAgentColumnConfig		

GetEnumerator

This method returns the enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

3.4.7 IAgentConfig

This interface makes the framework for the agent configuration available.

3.4.7.1 Properties

ColumnCount

This property contains the number of columns.

ColumnCount	
<i>int</i>	

ColumnList

This property contains a list with agent columns.

ColumnList	
<i>ColumnList</i>	

MaxColumnType

This property contains the maximum value of the EnumAgentColumnType enumerator.

This property has been retained for reasons of downward compatibility but is no longer required.

MaxColumnType	
<i>int</i>	

MediaType

This property contains the media type.

MediaType	
<i>EnumMediaType</i>	

NodeType

This property contains the node type.

NodeType	
<i>EnumNodeType</i>	

3.4.8 IAgentObject

This interface makes the framework for an agent available.

3.4.8.1 Properties

ACDContacts

This property contains a list with all active contacts of the agent:

ACDContacts	
<i> AcdContactList</i>	

AgentId

This property contains the AgentID:

AgentId	
<i>string</i>	

AgentIdentifier

This property contains the globally unique AgentID:

AgentIdentifier	
<i>string</i>	

AgentKey

This property contains the agent key:

AgentKey	
<i>string</i>	

AgentState (only OSCC V7):

This property contains the agent status:

AgentState	
<i>EnumAgentState</i>	

AgentTimeInState

This property contains the current duration of the present status in milliseconds.

AgentTimeInState	
string	

AveragePostProcessingTime (only OSCC V8)

This property contains the average post-processing time in seconds.

AveragePostProcessingTime	
<i>string</i>	

AverageTaskTime (only OSCC 7)

This property contains the average processing time in milliseconds.

AverageTaskTime	
<i>string</i>	

CustomValue1

This property is reserved for future customized implementations and is not yet in use.

CustomValue1	
<i>string</i>	

CustomValue10

This property is reserved for future customized implementations and is not yet in use.

CustomValue10	
<i>string</i>	

CustomValue2

This property is reserved for future customized implementations and is not yet in use.

CustomValue2	
<i>string</i>	

CustomValue3

This property is reserved for future customized implementations and is not yet in use.

CustomValue3	
<i>string</i>	

CustomValue4

This property is reserved for future customized implementations and is not yet in use.

CustomValue4	
<i>string</i>	

CustomValue5

This property is reserved for future customized implementations and is not yet in use.

CustomValue5	
<i>string</i>	

CustomValue6

This property is reserved for future customized implementations and is not yet in use.

CustomValue6	
<i>string</i>	

CustomValue7

This property is reserved for future customized implementations and is not yet in use.

CustomValue7	
<i>string</i>	

CustomValue8

This property is reserved for future customized implementations and is not yet in use.

CustomValue8	
<i>string</i>	

CustomValue9

This property is reserved for future customized implementations and is not yet in use.

CustomValue9	
<i>string</i>	

EligibleContacts

This property contains the number of contacts an agent can accept.

Note: The number of contacts is independant on the media (Voice, Callback, Email). This value only allows sensible content for agents owning only one mains kill (i.e. purely Voice Contact Center).

EligibleContacts	
<i>string</i>	

ExtensionNumber

This property contains the extension.

ExtensionNumber	
<i>string</i>	

Firstname

This property contains the first name.

Firstname	
<i>string</i>	

HandledTask

This property contains the number of calls handled.

HandledTask	
<i>string</i>	

LastName

This property contains the last name.

LastName	
<i>string</i>	

LoggedOnMedia

This property indicates the media for which the agent is logged on.

Note: effective as of V5.0.0.0.

LoggedOnMedia	
<i>EnumLoggedOnMedia</i>	

ReasonCode

This property contains the absence reason in legible text.

ReasonCode	
<i>string</i>	

Team

This property contains the team.

Team	
<i>ITeamEntry</i>	

TimeInUserPresenceState (only OSCC V8)

This property shows the time in the current state (presence) in seconds.

TimeInUserPresenceState	
<i>string</i>	

TotalTalkTime

This property indicates the absolute talk time of the agent.

Note: effective as of V5.0.0.0.

TotalTalkTime	
<i>string</i>	

UniquelIdentifier

This property contains a unique identifier.

UniquelIdentifier	
<i>string</i>	

UserPresenceState (only OSCC V8)

This property indicates the agent presence state.

UserPresenceState	
<i>EnumUserPresenceState</i>	

UserRoutingState (only OSCC V8)

This property indicates the agent routing state.

UserRoutingState	
<i>EnumUserRoutingState</i>	

UserRoutingStateReason (only OSCC V8)

This property indicates the agent routing state reason.

UserRoutingStateReason	
<i>string</i>	

Valid

This property indicates whether the Agent object is valid or invalid.

Valid	
<i>EnumValid</i>	

3.4.9 IAgentObjectList

This interface makes the model for a list of Agent objects available.

3.4.9.1 Properties

Count

This property contains the number of lists.

Count	
<i>int</i>	

3.4.9.2 Methods

get_Item

This method returns the Agent object.

get_Item		
<i>Parameters</i>		
Index	<i>int</i>	column index
<i>Return</i>		
IAgentObject		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

3.4.10 IAgentsData

This interface makes the framework for the agent data available.

3.4.10.1 Properties

AgentCount

This property contains the number of agents.

AgentCount	
<i>int</i>	

AgentObjectList

This property contains the AgentObject list.

AgentObjectList	
<i>IAgentObjectsList</i>	

FieldCount

This property contains the number of fields.

FieldCount	
<i>int</i>	

ImmediateXfer

This property contains yes or no; if OSCC or Genesys is used, always no. (1 = yes).

ImmediateXfer	
<i>bool</i>	

MediaType

This property contains the media type.

MediaType	
<i>EnumMediaType</i>	

NodeType

This property contains the node type.

NodeType	
<i>EnumNodeType</i>	

Valid

This property indicates whether the AgentData object is valid or invalid.

Valid	
<i>EnumValid</i>	

3.4.11 IColumnConfig

This interface makes the framework for the column configuration available.

3.4.11.1 Properties

ColumnIndex

This property contains the column index.

ColumnIndex	
<i>int</i>	

DataType

This property contains the data type.

DataType	
<i>EnumDataType</i>	

HeaderCustom

This property contains the customized header.

HeaderCustom	
<i>string</i>	

HeaderEnglish

This property contains the English header.

HeaderEnglish	
<i>string</i>	

HeaderFrance

This property contains the French header.

HeaderFrance	
<i>string</i>	

HeaderGerman

This property contains the German header.

HeaderGerman	
<i>string</i>	

HeaderSpain

This property contains the Spanish header.

HeaderSpain	
<i>string</i>	

MaxWidth

This property contains the maximum width of the column.

MaxWidth	
<i>int</i>	

RESERVED1

This property is reserved for future implementations and is not yet in use.

RESERVED1	
<i>string</i>	

RESERVED2

This property is reserved for future implementations and is not yet in use.

RESERVED2	
<i>string</i>	

RESERVED3

This property is reserved for future implementations and is not yet in use.

RESERVED3	
<i>string</i>	

RESERVED4

This property is reserved for future implementations and is not yet in use.

RESERVED4	
<i>string</i>	

RESERVED5

This property is reserved for future implementations and is not yet in use.

RESERVED5	
<i>string</i>	

3.4.12 IConfiguredViewsEvent

3.4.12.1 Properties

AgentCount

This property contains the number of agents.

AgentCount	
<i>int</i>	

AgentGroupCount

This property contains the number of agent groups.

AgentGroupCount	
<i>int</i>	

QueueCount

This property contains the number of queues.

QueueCount	
<i>int</i>	

3.4.13 IGroupColumnConfig

This interface makes the framework for the group column configuration available.

3.4.13.1 Properties

ColumnIndex

This property contains the column index.

ColumnIndex	
<i>int</i>	

DataType

This property contains the data type.

DataType	
<i>EnumDataType</i>	

HeaderCustom

This property contains the customized header.

HeaderCustom	
<i>string</i>	

HeaderEnglish

This property contains the English header.

HeaderEnglish	
<i>string</i>	

HeaderFrance

This property contains the French header.

HeaderFrance	
<i>string</i>	

HeaderGerman

This property contains the German header.

HeaderGerman	
<i>string</i>	

HeaderSpain

This property contains the Spanish header.

HeaderSpain	
<i>string</i>	

MaxWidth

This property contains the maximum width of the column.

MaxWidth	
<i>int</i>	

RESERVED1

This property is reserved for future implementations and is not yet in use.

RESERVED1	
<i>string</i>	

RESERVED2

This property is reserved for future implementations and is not yet in use.

RESERVED2	
<i>string</i>	

RESERVED3

This property is reserved for future implementations and is not yet in use.

RESERVED3	
<i>string</i>	

RESERVED4

This property is reserved for future implementations and is not yet in use.

RESERVED4	
<i>string</i>	

RESERVED5

This property is reserved for future implementations and is not yet in use.

RESERVED5	
<i>string</i>	

3.4.14 IGroupColumnConfigList

This interface makes the framework for a list of GroupColumnConfig objects available.

3.4.14.1 Properties

Count

This property contains the number of lists.

Count	
<i>int</i>	

3.4.14.2 Methods

get_Item

This method returns the group column configuration.

get_Item		
<i>Parameters</i>		
Index	<i>int</i>	column index
<i>Return</i>		
IGroupColumnConfig		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

3.4.15 IGroupConfig

This interface makes the framework for the group configuration available.

3.4.15.1 Properties

ColumnCount

This property contains the number of columns.

ColumnCount	
<i>int</i>	

ColumnList

This property contains a list with group configurations.

ColumnList	
ColumnList	

MaxColumnType

This property contains the maximum value of the EnumGroupColumnType enumerator.

This property has been retained for reasons of downward compatibility but is no longer required.

MaxColumnType	
<i>int</i>	

MediaType

This property contains the media type.

MediaType	
<i>EnumMediaType</i>	

NodeType

This property contains the node type.

NodeType	
<i>EnumNodeType</i>	

3.4.16 IGroupObject

This interface makes the framework for the Group object available.

3.4.16.1 Properties

CustomValue1

This property is reserved for future customized implementations and is not yet in use.

CustomValue1	
<i>string</i>	

CustomValue10

This property is reserved for future customized implementations and is not yet in use.

CustomValue10	
<i>string</i>	

CustomValue2

This property is reserved for future customized implementations and is not yet in use.

CustomValue2	
<i>string</i>	

CustomValue3

This property is reserved for future customized implementations and is not yet in use.

CustomValue3	
<i>string</i>	

CustomValue4

This property is reserved for future customized implementations and is not yet in use.

CustomValue4	
<i>string</i>	

CustomValue5

This property is reserved for future customized implementations and is not yet in use.

CustomValue5	
<i>string</i>	

CustomValue6

This property is reserved for future customized implementations and is not yet in use.

CustomValue6	
<i>string</i>	

CustomValue7

This property is reserved for future customized implementations and is not yet in use.

CustomValue7	
<i>string</i>	

CustomValue8

This property is reserved for future customized implementations and is not yet in use.

CustomValue8	
<i>string</i>	

CustomValue9

This property is reserved for future customized implementations and is not yet in use.

CustomValue9	
<i>string</i>	

GroupName

This property contains the group name.

GroupName	
<i>string</i>	

TotalAgentsAvailable

This property contains the number of agents in the "available" state.

TotalAgentsAvailable	
<i>string</i>	

TotalAgentsIdle

This property contains the number of agents in the "unavailable" state.

TotalAgentsIdle	
<i>string</i>	

TotalAgentsWrapUp

This property contains the number of agents in the "wrapup" state.

TotalAgentsWrapUp	
<i>string</i>	

TotalAgentsHandled

This property contains the number of agents in the "call" state.

TotalAgentsHandled	
<i>string</i>	

UniquelIdentifier

This property contains a unique identifier.

UniquelIdentifier	
<i>string</i>	

Valid

This property indicates whether the Group object is valid or invalid.

Valid	
<i>EnumValid</i>	

3.4.17 IGroupObjectList

This interface makes the framework for a list of GroupObjects available.

3.4.17.1 Properties

Count

This property contains the number of lists.

Count	
<i>int</i>	

3.4.17.2 Methods

get_Item

This method returns the IGroupObject object.

get_Item		
<i>Parameters</i>		
Index	<i>int</i>	column index
<i>Return</i>		
IGroupObject		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

3.4.18 IGroupData

This interface makes the framework for the group data available.

3.4.18.1 Properties

FieldCount

This property contains the number of fields.

FieldCount	
<i>int</i>	

GroupCount

This property contains the number of groups.

GroupCount	
<i>int</i>	

GroupObjectList

This property contains the GroupObjectList.

GroupObjectList	
<i>IGroupObjectsList</i>	

ImmediateXfer

This property contains yes or no; if OSCC or Genesys is used, always no. (1 = yes).

ImmediateXfer	
<i>bool</i>	

MediaType

This property contains the media type.

MediaType	
<i>EnumMediaType</i>	

NodeType

This property contains the node type.

NodeType	
<i>EnumNodeType</i>	

Valid

This property indicates whether the AgentData object is valid or invalid.

Valid	
<i>EnumValid</i>	

3.4.19 ILogoffResponse

This interface makes the framework for a Logoff Response object available.

3.4.19.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>EnumErrorCode</i>	

3.4.20 ILogonResponse

This interface makes the framework for a Logon Response object available.

3.4.20.1 Properties

Errorcode

This property contains the error code.

Errorcode	
<i>EnumErrorCode</i>	

3.4.21 IQueueColumnConfig

This interface makes the framework for the queue column configuration available.

3.4.21.1 Properties

ColumnIndex

This property contains the column index.

ColumnIndex	
<i>int</i>	

ColumnType

This property contains the column type.

ColumnType	
<i>EnumQueueColumnConfig</i>	

DataType

This property contains the data type.

DataType	
<i>EnumDataType</i>	

HeaderCustom

This property contains the customized header.

HeaderCustom	
<i>string</i>	

HeaderEnglish

This property contains the English header.

HeaderEnglish	
<i>string</i>	

HeaderFrance

This property contains the French header.

HeaderFrance	
<i>string</i>	

HeaderGerman

This property contains the German header.

HeaderGerman	
<i>string</i>	

HeaderSpain

This property contains the Spanish header.

HeaderSpain	
<i>string</i>	

MaxWidth

This property contains the maximum width of the column.

MaxWidth	
<i>int</i>	

RESERVED1

This property is reserved for future implementations and is not yet in use.

RESERVED1	
<i>string</i>	

RESERVED2

This property is reserved for future implementations and is not yet in use.

RESERVED2	
<i>string</i>	

RESERVED3

This property is reserved for future implementations and is not yet in use.

RESERVED3	
<i>string</i>	

RESERVED4

This property is reserved for future implementations and is not yet in use.

RESERVED4	
<i>string</i>	

RESERVED5

This property is reserved for future implementations and is not yet in use.

RESERVED5	
<i>string</i>	

3.4.22 IQueueColumnConfigList

This interface makes the framework for a list of QueueColumnConfig objects available.

3.4.22.1 Properties

Count

This property contains the number of lists.

Count	
<i>int</i>	

3.4.22.2 Methods

get_Item

This methods returns the QueueColumnConfig object.

get_Item		
<i>Parameters</i>		
Index	<i>int</i>	column index
<i>Return</i>		
IQueueColumnConfig		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

3.4.23 IQueueConfig

This interface makes the framework for the queue configuration available.

3.4.23.1 Properties

ColumnCount

This property contains the number of columns.

ColumnCount	
<i>int</i>	

ColumnList

This property contains a list with queue configurations.

ColumnList	
<i>ColumnList</i>	

MaxColumnType

This property contains the maximum value of the EnumQueueColumnType enumerator.

This property has been retained for reasons of downward compatibility but is no longer required.

MaxColumnType	
<i>int</i>	

MediaType

This property contains the media type.

MediaType	
<i>EnumMediaType</i>	

NodeType

This property contains the node type.

NodeType	
<i>EnumNodeType</i>	

3.4.24 IQueueObject

This interface makes the framework for a queue available.

3.4.24.1 Properties

AcqSteps

The property contains the average wait time for the queue in seconds.

AcqSteps	
<i> AcqStepList</i>	

AverageWaitTime

The property contains the average wait time for the queue in seconds.

AverageWaitTime	
<i>string</i>	

CustomValue1

This property is reserved for future customized implementations and is not yet in use.

CustomValue1	
<i>string</i>	

CustomValue10

This property is reserved for future customized implementations and is not yet in use.

CustomValue10	
<i>string</i>	

CustomValue2

This property is reserved for future customized implementations and is not yet in use.

CustomValue2	
<i>string</i>	

CustomValue3

This property is reserved for future customized implementations and is not yet in use.

CustomValue3	
<i>String</i>	

CustomValue4

This property is reserved for future customized implementations and is not yet in use.

CustomValue4	
<i>string</i>	

CustomValue5

This property is reserved for future customized implementations and is not yet in use.

CustomValue5	
<i>string</i>	

CustomValue6

This property is reserved for future customized implementations and is not yet in use.

CustomValue6	
<i>string</i>	

CustomValue7

This property is reserved for future customized implementations and is not yet in use.

CustomValue7	
<i>string</i>	

CustomValue8

This property is reserved for future customized implementations and is not yet in use.

CustomValue8	
<i>string</i>	

CustomValue9

This property is reserved for future customized implementations and is not yet in use.

CustomValue9	
<i>string</i>	

LongestItem

This property contains the duration in seconds of the contact in the queue that has been waiting the longest.

LongestItem	
<i>string</i>	

QueueAccessibility

This property contains the accessibility of the queue.

QueueAccessibility	
<i>string</i>	

QueueKey

The property contains the queue key.

QueueKey	
<i>string</i>	

QueueName

This property contains the queue name.

QueueName	
<i>string</i>	

ServiceLevel

This property contains the service level (there is only one main service level) of the queue.

ServiceLevel	
<i>string</i>	

TerminationRate

This property contains the termination rate of the queue.

TerminationRate	
<i>string</i>	

TotalAgentsPresenceIdle

This property contains the number of agents in presence state *Idle*. (only from OSCC V8 or higher)

TotalAgentsPresenceIdle	
<i>integer</i>	

TotalAgentsPresenceBusy

This property contains the number of agents in presence state *Busy*. (only from OSCC V8 or higher)

TotalAgentsPresenceBusy	
<i>integer</i>	

TotalAgentsPresenceAway

This property contains the number of agents in presence state *Away*. (only from OSCC V8 or higher)

TotalAgentsPresenceAway	
<i>integer</i>	

TotalAgentsPresenceActive

This property contains the number of agents in presence state *Active*. (only from OSCC V8 or higher)

TotalAgentsPresenceActive	
<i>integer</i>	

TotalDeflect

This property contains the number of overflowed calls in the queue.

TotalDeflect	
<i>string</i>	

TotalDisconnected

This property contains the number of disconnected calls in the queue.

TotalDisconnected	
<i>string</i>	

TotalDisconnectedInService

This property contains the number of disconnected calls in the queue in the service level interval.

TotalDisconnectedInService	
<i>string</i>	

TotalDisconnectedInService2

This property contains the number of disconnected calls in the queue in the service level interval 2.

TotalDisconnectedInService2	
<i>string</i>	

TotalDisconnectedInService3

This property contains the number of disconnected calls in the queue in the service level interval 3.

TotalDisconnectedInService3	
<i>string</i>	

TotalDisconnectedInService4

This property contains the number of disconnected calls in the queue in the service level interval 4.

TotalDisconnectedInService4	
<i>string</i>	

TotalDisconnectedInService5

This property contains the number of disconnected calls in the queue in the service level interval 5.

TotalDisconnectedInService5	
<i>string</i>	

TotalHandled

This property contains the number of answered calls in the queue.

TotalHandled	
<i>string</i>	

TotalHandledInService

This property contains the number of answered calls in the queue in the service level interval.

TotalHandledInService	
<i>string</i>	

TotalHandledInService2

This property contains the number of answered calls in the queue in the service level interval 2.

TotalHandledInService2	
<i>string</i>	

TotalHandledInService3

This property contains the number of answered calls in the queue in the service level interval 3.

TotalHandledInService3	
<i>string</i>	

TotalHandledInService4

This property contains the number of answered calls in the queue in the service level interval 4.

TotalHandledInService4	
<i>string</i>	

TotalHandledInService5

This property contains the number of answered calls in the queue in the service level interval 5.

TotalHandledInService5	
<i>string</i>	

TotalOffered

This property contains the number of received calls in the queue.

TotalOffered	
<i>string</i>	

UniquelIdentifier

This property contains a unique identifier.

UniquelIdentifier	
<i>string</i>	

Valid

This property indicates whether the Queue object is valid or invalid.

Valid	
<i>EnumValid</i>	

WaitingContacts

This property contains the number of callers currently waiting in the queue.

WaitingContacts	
<i>string</i>	

3.4.25 IQueueObjectsList

This interface makes the framework for a list of Queue objects available.

3.4.25.1 Properties

Count

This property contains the number of lists.

Count	
<i>int</i>	

3.4.25.2 Methods

get_Item

This method returns the IQueueObject object.

get_Item		
<i>Parameters</i>		
Index	<i>int</i>	column index
<i>Return</i>		
IQueueObject		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

3.4.26 IQueuesData

This interface makes the framework for the queue data available.

3.4.26.1 Properties

FieldCount

This property contains the number of fields.

FieldCount	
<i>int</i>	

ImmediateXfer

This property contains yes or no; if OSCC or Genesys is used, always no. (1 = yes).

ImmediateXfer	
<i>bool</i>	

MediaType

This property contains the media type.

MediaType	
<i>EnumMediaType</i>	

NodeType

This property contains the node type.

NodeType	
<i>EnumNodeType</i>	

QueueCount

This property contains the number of queues.

QueueCount	
<i>int</i>	

QueueObjectsList

This property contains the QueueObjects lists.

QueueObjectsList	
<i>IQueueObjectsList</i>	

Valid

This property indicates whether the QueueData object is valid or invalid.

Valid	
<i>EnumValid</i>	

3.4.27 IQueueVirtualAgentObject

This interface makes the framework for virtual agents available.

3.4.27.1 Properties

QueueKey

This property contains the queue key.

QueueKey	
<i>string</i>	

QueueName

This property contains the queue name.

QueueName	
<i>string</i>	

UniquelIdentifier

This property contains a unique identifier.

UniquelIdentifier	
<i>string</i>	

VirtualAgentsList

This property contains a VirtualAgent list.

VirtualAgentsList	
<i>IVirtualAgentObjectsList</i>	

3.4.28 IQueueVirtualAgentObjectsList

This interface makes the framework for a list of VirtualAgent objects available.

3.4.28.1 Properties

Count

This property contains the number of lists.

Count	
<i>int</i>	

get_Item

This method returns the IQueueVirtualAgentObject object.

get_Item		
<i>Parameters</i>		
Index	<i>int</i>	column index
<i>Return</i>		
IQueueVirtualAgentObject		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

3.4.29 ITeamConfig

This interface makes the framework for the Team Configuration object available.

3.4.29.1 Properties

MediaType

This property contains the media type.

MediaType	
<i>EnumMediaType</i>	

NodeType

This property contains the node type.

NodeType	
<i>EnumNodeType</i>	

TeamEntryCount

This property contains the number of teams.

TeamEntryCount	
<i>int</i>	

TeamList

This property contains the team list.

TeamList	
<i>ITeamList</i>	

3.4.30 ITeamEntry

This interface makes the framework for a Team object available.

3.4.30.1 Properties

Description

This property contains the description of the team.

Description	
<i>string</i>	

Key

This property contains the team key.

Key	
<i>string</i>	

Name

This property contains the team name.

Name	
<i>string</i>	

3.4.31 ITeamList

This interface makes a framework for a list of Team objects available.

3.4.31.1 Properties

Count

This property contains the number of lists.

Count	
<i>int</i>	

3.4.31.2 Methods

get_Item

This methods returns the ITeamEntry object.

get_Item		
<i>Parameters</i>		
Index	<i>int</i>	column index
<i>Return</i>		
ITeamEntry		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

3.4.32 IServerErrorEvent

3.4.32.1 Properties

ServerErrorEvent

This property contains the ServerErrorEvent.

ServerErrorEvent	
<i>EnumServerErrorType</i>	

3.4.33 IVirtualAgentObject

This interface makes the framework for VirtualAgent objects available.

3.4.33.1 Properties

AgentKey

This property contains the AgentKey.

AgentKey	
<i>string</i>	

Firstname

This property contains the agent's first name.

Firstname	
<i>string</i>	

LastName

This property contains the agent's last name.

LastName	
<i>string</i>	

3.4.34 IVirtualAgentObjectsList

This interface makes the framework for a list of VirtualAgent objects available.

3.4.34.1 Properties

Count

This property contains the number of lists.

Count	
<i>int</i>	

get_Item

This methods returns the VirtualAgent object.

get_Item		
<i>Parameters</i>		
Index	<i>int</i>	column index
<i>Return</i>		
IVirtualAgentObject		

GetEnumerator

This method returns an enumerator.

GetEnumerator		
<i>Return</i>		
IEnumerator		

3.4.35 IVirtualAgentsData

This interface makes the framework for the virtual agent data available.

3.4.35.1 Properties

MediaType

This property contains the media type.

MediaType	
<i>EnumMediaType</i>	

NodeType

This property contains the node type.

NodeType	
<i>EnumNodeType</i>	

QueueCount

This property contains the number of queues.

QueueCount	
<i>int</i>	

QueueList

This property contains a queue list.

QueueList	
<i>IQueueVirtualAgentsObjectsList</i>	