CSTA

CSTADLL
Version 2.1.2
CSTADLL
Reference Manual

Objective Systems, Inc. version 2.1.2 May 2015

Copyright Notice

Author's Contact Information

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

Contents

1 CSTADLL **1**

2 Namespace Documentation **3**

2.1 Package Com.Objsys.Csta.Common . . . 3

2.1.1 Detailed Description . . . 3

2.2 Package Com.Objsys.Csta.Devices . . . 4

2.2.1 Detailed Description . . . 4

2.3 Package Com.Objsys.Csta.Phase1 . . . 5

2.3.1 Detailed Description . . . 5

2.4 Package Com.Objsys.Csta.Phase2 . . . 6

2.4.1 Detailed Description . . . 6

2.5 Package Com.Objsys.Csta.Phase3 . . . 7

2.5.1 Detailed Description . . . 7

3 Class Documentation **9**

3.1 Alcatel4400 Class Reference . . . 9

3.1.1 Detailed Description . . . 9

3.1.2 Constructor & Destructor Documentation . . . 9

3.1.2.1 Alcatel4400 . . . 9

3.1.2.2 Alcatel4400 . . . 9

3.1.3 Member Function Documentation . . . 10

3.1.3.1 MakeACSEAssociation . . . 10

3.2 AlcatelOXE Class Reference . . . 11

3.2.1 Detailed Description . . . 11

3.2.2 Constructor & Destructor Documentation . . . 11

3.2.2.1 AlcatelOXE . . . 11

3.2.2.2 AlcatelOXE . . . 11

3.3 AlcatelOXO Class Reference . . . 12

3.3.1 Detailed Description . . . 12

3.3.2 Constructor & Destructor Documentation . . . 12
3.3.2.1 AlcatelOXO . . . 12
3.3.2.2 AlcatelOXO . . . 12
3.3.3 Member Function Documentation . . . 12
3.3.3.1 MakeACSEAssociation . . . 12
3.4 CSTAContext Class Reference . . . 13
3.4.1 Detailed Description . . . 13
3.4.2 Property Documentation . . . 13
3.4.2.1 ResponseFromPBX . . . 13
3.4.2.2 ResponsesFromPBX . . . 13
3.5 CSTAResponseInfo Class Reference . . . 14
3.5.1 Detailed Description . . . 14
3.5.2 Property Documentation . . . 14
3.5.2.1 ResponseFromPBX . . . 14
3.5.2.2 ResponsesFromPBX . . . 14
3.5.2.3 StatusCode . . . 14
3.5.2.4 StatusMessage . . . 14
3.6 GenericCSTAp1 Class Reference . . . 15
3.6.1 Detailed Description . . . 15
3.6.2 Constructor & Destructor Documentation . . . 16
3.6.2.1 GenericCSTAp1 . . . 16
3.6.2.2 GenericCSTAp1 . . . 16
3.6.3 Member Function Documentation . . . 16
3.6.3.1 AnswerCall . . . 16
3.6.3.2 AnswerCall . . . 16
3.6.3.3 AnswerCall . . . 16
3.6.3.4 ClearConnection . . . 17
3.6.3.5 ClearDoNotDisturb . . . 17
3.6.3.6 ClearMessageWaiting . . . 17
3.6.3.7 ConsultationCall . . . 17
3.6.3.8 DivertCall . . . 18
3.6.3.9 EncodeROSERequestHeader . . . 18
3.6.3.10 MakeACSEAssociation . . . 18
3.6.3.11 MakeCall . . . 18
3.6.3.12 MonitorStart . . . 19
3.6.3.13 MonitorStop . . . 19
3.6.3.14 MonitorStop . . . 19

3.6.3.15 MonitorStop . . . 19
3.6.3.16 QueryDevice . . . 20
3.6.3.17 ReleaseACSEAssociation . . . 20
3.6.3.18 RetrieveCall . . . 20
3.6.3.19 SetDoNotDisturb . . . 20
3.6.3.20 SetMessageWaiting . . . 20
3.6.3.21 SnapshotDevice . . . 21
3.6.3.22 TransferCall . . . 21
3.6.3.23 TransferCall . . . 21
3.6.4 Property Documentation . . . 21
3.6.4.1 SessionObject . . . 21
3.6.4.2 ThreadContext . . . 21
3.7 GenericCSTAp2 Class Reference . . . 22
3.7.1 Detailed Description . . . 22
3.7.2 Constructor & Destructor Documentation . . . 23
3.7.2.1 GenericCSTAp2 . . . 23
3.7.2.2 GenericCSTAp2 . . . 23
3.7.3 Member Function Documentation . . . 23
3.7.3.1 AnswerCall . . . 23
3.7.3.2 AnswerCall . . . 23
3.7.3.3 AnswerCall . . . 23
3.7.3.4 ClearConnection . . . 24
3.7.3.5 ClearDoNotDisturb . . . 24
3.7.3.6 ClearMessageWaiting . . . 24
3.7.3.7 ConsultationCall . . . 24
3.7.3.8 DivertCall . . . 25
3.7.3.9 EncodeROSERequestHeader . . . 25
3.7.3.10 MakeACSEAssociation . . . 25
3.7.3.11 MakeCall . . . 25
3.7.3.12 MonitorStart . . . 26
3.7.3.13 MonitorStop . . . 26
3.7.3.14 MonitorStop . . . 26
3.7.3.15 MonitorStop . . . 26
3.7.3.16 QueryDevice . . . 27
3.7.3.17 ReleaseACSEAssociation . . . 27
3.7.3.18 RetrieveCall . . . 27
3.7.3.19 SetDoNotDisturb . . . 27

3.7.3.20 SetMessageWaiting . . . 27
3.7.3.21 SnapshotDevice . . . 28
3.7.3.22 TransferCall . . . 28
3.7.3.23 TransferCall . . . 28
3.7.4 Property Documentation . . . 28
3.7.4.1 SessionObject . . . 28
3.7.4.2 ThreadContext . . . 28
3.8 GenericCSTAp3 Class Reference . . . 29
3.8.1 Detailed Description . . . 30
3.8.2 Constructor & Destructor Documentation . . . 30
3.8.2.1 GenericCSTAp3 . . . 30
3.8.2.2 GenericCSTAp3 . . . 30
3.8.3 Member Function Documentation . . . 30
3.8.3.1 AnswerCall . . . 30
3.8.3.2 AnswerCall . . . 30
3.8.3.3 AnswerCall . . . 31
3.8.3.4 ClearConnection . . . 31
3.8.3.5 ClearDoNotDisturb . . . 31
3.8.3.6 ClearMessageWaiting . . . 31
3.8.3.7 ConsultationCall . . . 31
3.8.3.8 EncodeROSERequestHeader . . . 32
3.8.3.9 GetAgentState . . . 32
3.8.3.10 GetDoNotDisturb . . . 32
3.8.3.11 GetSFDevices . . . 33
3.8.3.12 GetSFDevices . . . 33
3.8.3.13 HoldCall . . . 33
3.8.3.14 MakeACSEAssociation . . . 33
3.8.3.15 MakeCall . . . 33
3.8.3.16 MonitorStart . . . 34
3.8.3.17 MonitorStop . . . 34
3.8.3.18 MonitorStop . . . 34
3.8.3.19 MonitorStop . . . 34
3.8.3.20 ReleaseACSEAssociation . . . 35
3.8.3.21 RetrieveCall . . . 35
3.8.3.22 RingDevice . . . 35
3.8.3.23 SendData . . . 35
3.8.3.24 SetDisplay . . . 36

3.8.3.25 SetDoNotDisturb . . . 36
3.8.3.26 SetMessageWaiting . . . 36
3.8.3.27 SingleStepTransfer . . . 36
3.8.3.28 SingleStepTransfer . . . 37
3.8.3.29 SnapshotCall . . . 37
3.8.3.30 SnapshotDevice . . . 37
3.8.3.31 StartDataPath . . . 37
3.8.3.32 StopDataPath . . . 38
3.8.3.33 StopRing . . . 38
3.8.3.34 TransferCall . . . 38
3.8.3.35 TransferCall . . . 38
3.8.4 Property Documentation . . . 39
3.8.4.1 SessionObject . . . 39
3.8.4.2 ThreadContext . . . 39
3.9 IETF_CSTAp1 Class Reference . . . 40
3.9.1 Detailed Description . . . 40
3.9.2 Constructor & Destructor Documentation . . . 40
3.9.2.1 IETF_CSTAp1 . . . 40
3.9.2.2 IETF_CSTAp1 . . . 40
3.10 IETF_CSTAp2 Class Reference . . . 41
3.10.1 Detailed Description . . . 41
3.10.2 Constructor & Destructor Documentation . . . 41
3.10.2.1 IETF_CSTAp2 . . . 41
3.10.2.2 IETF_CSTAp2 . . . 41
3.11 IETF_CSTAp3 Class Reference . . . 42
3.11.1 Detailed Description . . . 42
3.11.2 Constructor & Destructor Documentation . . . 42
3.11.2.1 IETF_CSTAp3 . . . 42
3.11.2.2 IETF_CSTAp3 . . . 42
3.12 LicenseException Class Reference . . . 43
3.12.1 Detailed Description . . . 43
3.13 PanasonicKXTDA Class Reference . . . 44
3.13.1 Detailed Description . . . 44
3.13.2 Constructor & Destructor Documentation . . . 44
3.13.2.1 PanasonicKXTDA . . . 44
3.13.2.2 PanasonicKXTDA . . . 44
3.14 PanasonicKXTDE Class Reference . . . 45

3.14.1 Detailed Description . . . 45
3.14.2 Member Enumeration Documentation . . . 45
3.14.2.1 DeviceDataTypes . . . 45
3.14.3 Constructor & Destructor Documentation . . . 45
3.14.3.1 PanasonicKXTDE . . . 45
3.14.3.2 PanasonicKXTDE . . . 46
3.14.4 Member Function Documentation . . . 46
3.14.4.1 AcquireControlRight . . . 46
3.14.4.2 ClearMessageWaiting . . . 46
3.14.4.3 GetDeviceData . . . 46
3.14.4.4 GetGroupMembers . . . 47
3.14.4.5 GetSFDevices . . . 47
3.14.4.6 PDFStart . . . 47
3.14.4.7 PDFStop . . . 47
3.14.4.8 ReleaseControlRight . . . 48
3.14.4.9 ResetDisplay . . . 48
3.14.4.10 SendKmeMessage . . . 48
3.14.4.11 SetMessageWaiting . . . 48
3.15 PanasonicNCP Class Reference . . . 49
3.15.1 Detailed Description . . . 49
3.15.2 Constructor & Destructor Documentation . . . 49
3.15.2.1 PanasonicNCP . . . 49
3.15.2.2 PanasonicNCP . . . 49
3.16 PBXSession Class Reference . . . 50
3.16.1 Detailed Description . . . 50
3.16.2 Constructor & Destructor Documentation . . . 50
3.16.2.1 PBXSession . . . 50
3.16.3 Member Function Documentation . . . 51
3.16.3.1 AsyncCallback . . . 51
3.16.3.2 Close . . . 51
3.16.3.3 ConnectionCallback . . . 51
3.16.3.4 Open . . . 51
3.16.3.5 SendACSEMessage . . . 51
3.16.3.6 SendMessage . . . 52
3.16.3.7 SendMessage . . . 52
3.16.3.8 WaitForROSEResponse . . . 52
3.16.4 Property Documentation . . . 52

3.16.4.1 ClientCallback . . . 52
3.16.4.2 ConnectionLostCallback . . . 52
3.16.4.3 DebugClientCallback . . . 53
3.16.4.4 DebugMode . . . 53
3.16.4.5 MaxReceiveTimeout . . . 53
3.16.4.6 PBXSystem . . . 53
3.16.4.7 Port . . . 53
3.17 PBXSessionException Class Reference . . . 54
3.17.1 Detailed Description . . . 54
3.18 PBXSessionHelper Class Reference . . . 55
3.18.1 Detailed Description . . . 55
3.18.2 Property Documentation . . . 55
3.18.2.1 LoggingEnabled . . . 55
3.18.2.2 LoggingFolder . . . 55
3.19 Phase1Opcodes Class Reference . . . 56
3.19.1 Detailed Description . . . 56
3.19.2 Member Enumeration Documentation . . . 56
3.19.2.1 Opcodes . . . 56
3.20 Phase2Opcodes Class Reference . . . 57
3.20.1 Detailed Description . . . 57
3.20.2 Member Enumeration Documentation . . . 57
3.20.2.1 Opcodes . . . 57
3.21 Phase3Opcodes Class Reference . . . 58
3.21.1 Detailed Description . . . 58
3.21.2 Member Enumeration Documentation . . . 58
3.21.2.1 Opcodes . . . 58
3.22 PhilipsSopho Class Reference . . . 59
3.22.1 Detailed Description . . . 59
3.22.2 Constructor & Destructor Documentation . . . 59
3.22.2.1 PhilipsSopho . . . 59
3.22.2.2 PhilipsSopho . . . 59
3.22.3 Member Function Documentation . . . 59
3.22.3.1 MakeACSEAssociation . . . 59
3.22.3.2 MakeACSEAssociation . . . 60
3.23 ROSEParseInfo Class Reference . . . 61
3.23.1 Detailed Description . . . 61
3.24 SiemensCap Class Reference . . . 62

3.24.1 Detailed Description . . . 62
3.24.2 Constructor & Destructor Documentation . . . 62
3.24.2.1 SiemensCap . . . 62
3.24.2.2 SiemensCap . . . 62
3.25 SiemensHicom300 Class Reference . . . 63
3.25.1 Detailed Description . . . 63
3.25.2 Constructor & Destructor Documentation . . . 63
3.25.2.1 SiemensHicom300 . . . 63
3.25.2.2 SiemensHicom300 . . . 63
3.26 SiemensHipath3000p2 Class Reference . . . 64
3.26.1 Detailed Description . . . 64
3.26.2 Constructor & Destructor Documentation . . . 64
3.26.2.1 SiemensHipath3000p2 . . . 64
3.26.2.2 SiemensHipath3000p2 . . . 64
3.27 SiemensHipath3000p3 Class Reference . . . 65
3.27.1 Detailed Description . . . 65
3.27.2 Constructor & Destructor Documentation . . . 65
3.27.2.1 SiemensHipath3000p3 . . . 65
3.27.2.2 SiemensHipath3000p3 . . . 65
3.27.3 Member Function Documentation . . . 65
3.27.3.1 MakeACSEAssociation . . . 65
3.28 SiemensHipath4000 Class Reference . . . 66
3.28.1 Detailed Description . . . 66
3.28.2 Constructor & Destructor Documentation . . . 66
3.28.2.1 SiemensHipath4000 . . . 66
3.28.2.2 SiemensHipath4000 . . . 66
3.29 SiemensRealitis Class Reference . . . 67
3.29.1 Detailed Description . . . 67
3.29.2 Constructor & Destructor Documentation . . . 67
3.29.2.1 SiemensRealitis . . . 67
3.29.2.2 SiemensRealitis . . . 67
3.30 SingleStepTransferInfo Class Reference . . . 68
3.30.1 Detailed Description . . . 68
3.30.2 Property Documentation . . . 68
3.30.2.1 TransferFromDevice . . . 68
3.30.2.2 TransferringCallID . . . 68
3.30.2.3 TransferToDevice . . . 68

3.31 SocketState Class Reference . . . 69
3.31.1 Detailed Description . . . 69
3.31.2 Property Documentation . . . 69
3.31.2.1 AckBuffer . . . 69
3.31.2.2 ReadBuffer . . . 69
3.31.2.3 ReadBuffers . . . 69
3.31.2.4 TotalLength . . . 69
3.32 TadiranCoral Class Reference . . . 70
3.32.1 Detailed Description . . . 70
3.32.2 Constructor & Destructor Documentation . . . 70
3.32.2.1 TadiranCoral . . . 70
3.32.2.2 TadiranCoral . . . 70
3.33 TransferCallInfo Class Reference . . . 71
3.33.1 Detailed Description . . . 71
3.33.2 Property Documentation . . . 71
3.33.2.1 ActiveDeviceCallID . . . 71
3.33.2.2 ActiveDeviceNumber . . . 71
3.33.2.3 ConnectedDevice . . . 71
3.33.2.4 HeldDevice . . . 71
3.33.2.5 HeldDeviceCallID . . . 71
3.33.2.6 HeldDeviceNumber . . . 71
3.34 UnifyOpenscapeX5 Class Reference . . . 72
3.34.1 Detailed Description . . . 72
3.34.2 Constructor & Destructor Documentation . . . 72
3.34.2.1 UnifyOpenscapeX5 . . . 72
3.34.2.2 UnifyOpenscapeX5 . . . 72

Chapter 1

CSTADLL

The CSTADLL product is a Microsoft .NET 4.0 DLL that allows client code to use a single library to communicate with a PBX device, regardless of what CSTA phase that device uses. A version of the DLL built with .NET 3.5 is also provided in the kit.

The DLL uses the following namespaces:

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

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

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

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

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

The DLL allows a client to have a session with a single PBX system, during which the client can send ACSE and CSTA messages to the PBX and receive responses.

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

The CSTADLL kit includes several samples to guide you in writing your own code. The name of the sample conveys some information about the sample. If the name of the sample starts with DLL, it means the sample shows how to use one of the helper methods that the DLL exposes. If the name of the sample starts with CSTA, then that sample still shows a way to make use of the DLL, but instead of using one of the DLL's helper methods, the sample instead shows how to use the DLL to send to a PBX a CSTA message for which the DLL doesn't expose a helper method.

After either DLL or CSTA in the sample name is an indicator of what language the sample is written in. Cs is used to indicate that the sample is written in C#. Vb will be used to indicate that the sample is written in Visual BASIC.

Similar codes will be used if samples are provided in other languages; e.g., perhaps CppCLI for samples written in C++/CLI.

For example, the sample DLLCsAnswerCall is a sample that shows how to use one of the DLL's helper methods in C# to instruct a PBX to answer a call. The sample CSTACsButtonPress shows how to use the DLL with C# to send a PBX the CSTA ButtonPress message, for which there is no exposed helper method.

The classes and methods exposed by the all-phases DLL within the CSTADLL package are probably sufficient to handle ACSE and CSTA operations for most PBX devices. But if needed, you can write a class of your own to handle ACSE and CSTA operations for a PBX device that the CSTADLL software doesn't explicitly support. The sample DLLCsNewPBX shows how this might be accomplished. This sample contains code for a small separate DLL that could be used to support a fictitious PBX device called the AwesomePBX100. The assumption in the sample is that this device uses standard phase 2 messages for all operations except for the ACSE make association message. This message is the one message that is most commonly different from one PBX to the next. The DLLCsNewPBX sample shows how the MakeACSEAssociation method within the GenericCSTAp2 class (the lowest level base class for this sample) can be overridden in a class that you can write. The override implementation handles the details that are specific to the device. Other methods within GenericCSTAp2 could also be overridden as needed.

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

Chapter 2

Namespace Documentation

2.1 Package Com.Objsys.Csta.Common

Classes

- class CSTAContext
- class CSTAResponseInfo
- class LicenseException
- class PBXSession
- class PBXSessionException
- class PBXSessionHelper
- class ROSEParseInfo
- class SocketState

2.1.1 Detailed Description

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

2.2 Package Com.Objsys.Csta.Devices

Classes

- class Alcatel4400
- class AlcatelOXE
- class AlcatelOXO
- class PanasonicKXTDA
- class PanasonicKXTDE
- class PanasonicNCP
- class PhilipsSopho
- class SiemensCap
- class SiemensHicom300
- class SiemensHipath3000p2
- class SiemensHipath3000p3
- class SiemensHipath4000
- class SiemensRealitis
- class TadiranCoral
- class UnifyOpenscapeX5

2.2.1 Detailed Description

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

2.3 Package Com.Objsys.Csta.Phase1

Classes

- class GenericCSTAp1
- class IETF_CSTAp1
- class Phase1Opcodes

2.3.1 Detailed Description

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

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

2.4 Package Com.Objsys.Csta.Phase2

Classes

- class GenericCSTAp2
- class IETF_CSTAp2
- class Phase2Opcodes

2.4.1 Detailed Description

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

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

2.5 Package Com.Objsys.Csta.Phase3

Classes

- class GenericCSTAp3
- class IETF_CSTAp3
- class Phase3Opcodes
- class SingleStepTransferInfo
- class TransferCallInfo

2.5.1 Detailed Description

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

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

Chapter 3

Class Documentation

3.1 Alcatel4400 Class Reference

Inherits Com::Objsys::Csta::Phase2::IETF_CSTAp2.

Inherited by AlcatelOXE.

Public Member Functions

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

3.1.1 Detailed Description

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

3.1.2 Constructor & Destructor Documentation

3.1.2.1 Alcatel4400 (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.1.2.2 Alcatel4400 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.1.3 Member Function Documentation

3.1.3.1 override CSTAResponseInfo MakeACSEAssociation () `[virtual]`

Establish an ACSE association with the PBX.

Returns

A CSTAResponseInfo object.

Reimplemented from GenericCSTAp2.

3.2 AlcatelOXE Class Reference

Inherits Com::Objsys::Csta::Devices::Alcatel4400.

Public Member Functions

- AlcatelOXE (PBXSession sessionObject)
- AlcatelOXE (string pbxSystem, int port)

3.2.1 Detailed Description

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

3.2.2 Constructor & Destructor Documentation

3.2.2.1 AlcatelOXE (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.2.2.2 AlcatelOXE (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.3 AlcatelOXO Class Reference

Inherits Com::Objsys::Csta::Phase1::IETF_CSTAp1.

Public Member Functions

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

3.3.1 Detailed Description

Implements CSTA phase 1 operations for the Alcatel OXO device.

3.3.2 Constructor & Destructor Documentation

3.3.2.1 AlcatelOXO (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.3.2.2 AlcatelOXO (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.3.3 Member Function Documentation

3.3.3.1 override CSTAResponseInfo MakeACSEAssociation () `[virtual]`

Establish an ACSE association with the PBX.

Returns

A CSTAResponseInfo object.

Reimplemented from GenericCSTAp1.

3.4 CSTAContext Class Reference

Properties

- byte[ ] ResponseFromPBX `[get, set]`
- List< byte[ ]> ResponsesFromPBX `[get, set]`

3.4.1 Detailed Description

The CSTAContext class contains information needed to manage the interaction between the thread and the PBX.

3.4.2 Property Documentation

3.4.2.1 byte [ ] ResponseFromPBX `[get, set]`

See documentation for CSTAResponseInfo.ResponseFromPBX.

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

See documentation for CSTAResponseInfo.ResponsesFromPBX.

3.5 CSTAResponseInfo Class Reference

Properties

- byte[ ] ResponseFromPBX `[get, set]`
- List< byte[ ]> ResponsesFromPBX `[get, set]`
- int StatusCode `[get, set]`
- string StatusMessage `[get, set]`

3.5.1 Detailed Description

Contains information about a PBX operation that was attempted.

3.5.2 Property Documentation

3.5.2.1 byte [ ] ResponseFromPBX `[get, set]`

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

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

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

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

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

3.5.2.3 int StatusCode `[get, set]`

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

3.5.2.4 string StatusMessage `[get, set]`

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

3.6 GenericCSTAp1 Class Reference

Inherited by SiemensHicom300, and IETF_CSTAp1.

Public Member Functions

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

Properties

- PBXSession SessionObject `[get]`
- CSTAContext ThreadContext `[get]`

3.6.1 Detailed Description

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

3.6.2 Constructor & Destructor Documentation

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

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.6.2.2 GenericCSTAp1 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.6.3 Member Function Documentation

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

Answers a call.

Parameters

callToAnswer The connection id of the call to answer.

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

Returns

A CSTAResponseInfo object.

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

Answers a call.

Parameters

callToAnswer The ConnectionID of the call to answer.

Returns

A CSTAResponseInfo object.

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

Answers a call.

Parameters

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

Returns

A CSTAResponseInfo object.

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

Clears a connection.

Parameters

connToClear The ConnectionID of the connection to clear.

Returns

A CSTAResponseInfo object.

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

Turns off the Do Not Disturb functionality for a phone.

Parameters

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

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

Returns

A CSTAResponseInfo object.

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

Instruct the PBX to do a consultation call.

Parameters

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

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

Returns

A CSTAResponseInfo object.

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

Diverts a call from a source to a destination.

Parameters

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

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

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

encodeBuffer An Asn1BerEncodeBuffer instance containing the already encoded CSTA message.

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

Returns

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

3.6.3.10 virtual CSTAResponseInfo MakeACSEAssociation () [virtual]

Establish an ACSE association with the PBX.

Returns

A CSTAResponseInfo object.

Reimplemented in AlcatelOXO, and PhilipsSopho.

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

Instruct the PBX to place a call.

Parameters

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

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

Returns

A CSTAResponseInfo object.

3.6.3.12 virtual CSTAResponseInfo MonitorStart (string *deviceToMonitor*) `[virtual]`

Issues a MonitorStart request to the PBX.

Parameters

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

Returns

A CSTAResponseInfo object.

3.6.3.13 virtual CSTAResponseInfo MonitorStop (string *monitoredDevice*) `[virtual]`

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

Parameters

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

Returns

If no problems are encountered, the method returns a CSTAResponseInfo object containing the response from the PBX for the LAST Monitor Stop message.
If any problems are encountered, the method returns a CSTAResponseInfo object containing information about the error, including any response from the PBX for the problematic Monitor Stop message.

3.6.3.14 virtual CSTAResponseInfo MonitorStop (MonitorCrossRefID *crossRefID*) `[virtual]`

Stop a previously started PBX monitor request.

Parameters

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

Returns

A CSTAResponseInfo object.

3.6.3.15 virtual CSTAResponseInfo MonitorStop (int *crossRefInt*) `[virtual]`

Stop a previously started PBX monitor request.

Parameters

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

Returns

A CSTAResponseInfo object.

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

Queries a device.

Parameters

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

Returns

A CSTAResponseInfo object.

3.6.3.17 virtual CSTAResponseInfo ReleaseACSEAssociation () [virtual]

Releases an ACSE association with a PBX device.

Returns

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

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

Retrieves a held call.

Parameters

callToRetrieve The ConnectionID of the call to retrieve.

Returns

A CSTAResponseInfo object.

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

Sets the Do Not Disturb feature for a phone.

Parameters

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

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

Returns

A CSTAResponseInfo object.

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

Transfers a call from one device to another.

Parameters

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

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

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

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

Returns

A CSTAResponseInfo object.

3.6.4 Property Documentation

3.6.4.1 PBXSession SessionObject [get]

The PBXSession object associated with this instance.

3.6.4.2 CSTAContext ThreadContext [get]

The CSTAContext structure for this thread.

3.7 GenericCSTAp2 Class Reference

Inherited by SiemensHipath3000p2, and IETF_CSTAp2.

Public Member Functions

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

Properties

- PBXSession SessionObject `[get]`
- CSTAContext ThreadContext `[get]`

3.7.1 Detailed Description

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

3.7.2 Constructor & Destructor Documentation

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

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.7.2.2 GenericCSTAp2 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.7.3 Member Function Documentation

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

Answers a call.

Parameters

callToAnswer The connection id of the call to answer.

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

Returns

A CSTAResponseInfo object.

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

Answers a call.

Parameters

callToAnswer The ConnectionID of the call to answer.

Returns

A CSTAResponseInfo object.

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

Answers a call.

Parameters

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

Returns

A CSTAResponseInfo object.

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

Clears a connection.

Parameters

connToClear The ConnectionID of the connection to clear.

Returns

A CSTAResponseInfo object.

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

Turns off the Do Not Disturb functionality for a phone.

Parameters

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

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

Returns

A CSTAResponseInfo object.

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

Instruct the PBX to do a consultation call.

Parameters

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

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

Returns

A CSTAResponseInfo object.

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

Diverts a call from a source to a destination.

Parameters

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

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

Returns

A CSTAResponseInfo object.

3.7.3.9 int EncodeROSERequestHeader (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, Phase2Opcodes.Opcodes *opcode*)

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

Parameters

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

encodeBuffer An Asn1BerEncodeBuffer instance containing the already encoded CSTA message.

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

Returns

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

3.7.3.10 virtual CSTAResponseInfo MakeACSEAssociation () [virtual]

Establish an ACSE association with the PBX.

Returns

A CSTAResponseInfo object.

Reimplemented in Alcatel4400.

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

Instruct the PBX to place a call.

Parameters

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

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

Returns

A CSTAResponseInfo object.

3.7.3.12 virtual CSTAResponseInfo MonitorStart (string *deviceToMonitor*) `[virtual]`

Issues a MonitorStart request to the PBX.

Parameters

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

Returns

A CSTAResponseInfo object.

3.7.3.13 virtual CSTAResponseInfo MonitorStop (string *monitoredDevice*) `[virtual]`

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

Parameters

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

Returns

If no problems are encountered, the method returns a CSTAResponseInfo object containing the response from the PBX for the LAST Monitor Stop message.
If any problems are encountered, the method returns a CSTAResponseInfo object containing information about the error, including any response from the PBX for the problematic Monitor Stop message.

3.7.3.14 virtual CSTAResponseInfo MonitorStop (MonitorCrossRefID *crossRefID*) `[virtual]`

Stop a previously started PBX monitor request.

Parameters

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

Returns

A CSTAResponseInfo object.

3.7.3.15 virtual CSTAResponseInfo MonitorStop (int *crossRefInt*) `[virtual]`

Stop a previously started PBX monitor request.

Parameters

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

Returns

A CSTAResponseInfo object.

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

Queries a device.

Parameters

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

Returns

A CSTAResponseInfo object.

3.7.3.17 virtual CSTAResponseInfo ReleaseACSEAssociation () [virtual]

Releases an ACSE association with a PBX device.

Returns

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

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

Retrieves a held call.

Parameters

callToRetrieve The ConnectionID of the call to retrieve.

Returns

A CSTAResponseInfo object.

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

Sets the Do Not Disturb feature for a phone.

Parameters

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

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

Returns

A CSTAResponseInfo object.

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

Transfers a call from one device to another.

Parameters

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

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

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

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

Returns

A CSTAResponseInfo object.

3.7.4 Property Documentation

3.7.4.1 PBXSession SessionObject [get]

The PBXSession object associated with this instance.

3.7.4.2 CSTAContext ThreadContext [get]

The CSTAContext structure for this thread.

3.8 GenericCSTAp3 Class Reference

Inherited by SiemensHipath3000p3, and IETF_CSTAp3.

Public Member Functions

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

Properties

- PBXSession SessionObject `[get]`
- CSTAContext ThreadContext `[get]`

3.8.1 Detailed Description

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

3.8.2 Constructor & Destructor Documentation

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

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.8.2.2 GenericCSTAp3 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.8.3 Member Function Documentation

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

Answers a call.

Parameters

callToAnswer ConnectionID of an existing call (such as initiated through MakeCall()).

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

Returns

A CSTAResponseInfo object.

3.8.3.2 virtual CSTAResponseInfo AnswerCall (ConnectionID *callToAnswer*) `[virtual]`

Answers a call.

Parameters

callToAnswer The ConnectionID of the call to answer.

Returns

A CSTAResponseInfo object.

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

Answers a call.

Parameters

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

Returns

A CSTAResponseInfo object.

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

Clears a connection.

Parameters

connToClear The ConnectionID of the connection to clear.

Returns

A CSTAResponseInfo object.

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

Turns off the Do Not Disturb functionality for a phone.

Parameters

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

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

Returns

A CSTAResponseInfo object.

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

Instruct the PBX to do a consultation call.

Parameters

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

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

Returns

A CSTAResponseInfo object.

3.8.3.8 virtual int EncodeROSERequestHeader (CSTAResponseInfo *response*, Asn1BerEncodeBuffer *encodeBuffer*, Phase3Opcodes.Opcodes *opcode*) [virtual]

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

Parameters

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

encodeBuffer An Asn1BerEncodeBuffer instance containing the already encoded CSTA message.

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

Returns

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

3.8.3.9 virtual CSTAResponseInfo GetAgentState (string *targetDevice*) [virtual]

Gets the state of the agent associated with a device.

Parameters

targetDevice The device whose agent state is desired.

Returns

A CSTAResponseInfo object.

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

Gets the Do Not Disturb setting for a phone.

Parameters

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

Returns

A CSTAResponseInfo object.

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

Sends a Get Switching Function Devices request to the PBX.

Parameters

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

Returns

A CSTAResponseInfo object.

3.8.3.12 virtual CSTAResponseInfo GetSFDevices () [virtual]

Sends a Get Switching Function Devices request to the PBX.

Returns

A CSTAResponseInfo object.

Reimplemented in PanasonicKXTDE.

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

Instruct the PBX to hold a call.

Parameters

callToHold The ConnectionID of the call to be held.

Returns

A CSTAResponseInfo object.

3.8.3.14 virtual CSTAResponseInfo MakeACSEAssociation () [virtual]

Establish an ACSE association with the PBX.

Returns

A CSTAResponseInfo object.

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

Instruct the PBX to place a call.

Parameters

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

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

Returns

A CSTAResponseInfo object.

3.8.3.16 virtual CSTAResponseInfo MonitorStart (string *deviceToMonitor*) `[virtual]`

Issues a MonitorStart request to the PBX.

Parameters

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

Returns

A CSTAResponseInfo object.

3.8.3.17 virtual CSTAResponseInfo MonitorStop (string *monitoredDevice*) `[virtual]`

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

Parameters

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

Returns

If no problems are encountered, the method returns a CSTAResponseInfo object containing the response from the PBX for the LAST Monitor Stop message.
If any problems are encountered, the method returns a CSTAResponseInfo object containing information about the error, including any response from the PBX for the problematic Monitor Stop message.

3.8.3.18 virtual CSTAResponseInfo MonitorStop (MonitorCrossRefID *crossRefID*) `[virtual]`

Stop a previously started PBX monitor request.

Parameters

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

Returns

A CSTAResponseInfo object.

3.8.3.19 virtual CSTAResponseInfo MonitorStop (int *crossRefInt*) `[virtual]`

Stop a previously started PBX monitor request.

Parameters

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

Returns

A CSTAResponseInfo object.

3.8.3.20 virtual CSTAResponseInfo ReleaseACSEAssociation () [virtual]

Releases an ACSE association with a PBX device.

Returns

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

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

Retrieves a held call.

Parameters

callToRetrieve The ConnectionID of the call to retrieve.

Returns

A CSTAResponseInfo object.

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

Causes a telephony device to ring.

Parameters

targetDevice The device to ring.

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

ringPattern The indicator of the ring pattern to use.

Returns

A CSTAResponseInfo object.

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

Sends a text message to a telephony device.

Parameters

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

text The text to send to the telephony device.

Returns

A CSTAResponseInfo object.

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

Sends text to a telephony device's display

Parameters

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

text The text to be sent.

Returns

A CSTAResponseInfo object.

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

Sets the Do Not Disturb feature for a phone.

Parameters

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

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

Returns

A CSTAResponseInfo object.

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

Perform a single step transfer.

Parameters

sstInfo A SingleStepTransferInfo object.

Returns

A CSTAResponseInfo object.

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

Perform a single step transfer.

Parameters

callToTransfer The ConnectionID of the call to transfer.

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

Returns

A CSTAResponseInfo object.

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

Instruct the PBX to take a snapshot of a call.

Parameters

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

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

Returns

A CSTAResponseInfo object.

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

Opens up a data path to a specified device.

Parameters

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

Returns

A CSTAResponseInfo object.

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

Stops a previously established data path

Parameters

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

Returns

A CSTAResponseInfo object.

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

Stops a ringer on a telephony device.

Parameters

targetDevice The device for which the ringer is to stop.

targetRinger The id of the ringer to stop. This argument can be specified as a character string (e.g, "abc"), a hex string (e.g, "'010A05'H"), or a binary string (e.g, "'00000001000010100000101'B").

ringPattern The indicator of the ring pattern to stop.

Returns

A CSTAResponseInfo object.

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

Transfers a call from one device to another.

Parameters

tcInfo A TransferCallInfo object.

Returns

A CSTAResponseInfo object.

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

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

Parameters

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

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

Returns

A CSTAResponseInfo object.

3.8.4 Property Documentation

3.8.4.1 PBXSession SessionObject `[get]`

The PBXSession object associated with this instance.

3.8.4.2 CSTAContext ThreadContext `[get]`

The CSTAContext structure for this thread.

3.9 IETF_CSTAp1 Class Reference

Inherits Com::Objsys::Csta::Phase1::GenericCSTAp1.

Inherited by AlcatelOXO, PhilipsSopho, SiemensRealitis, and TadiranCoral.

Public Member Functions

- IETF_CSTAp1 (PBXSession sessionObject)
- IETF_CSTAp1 (string pbxSystem, int port)

3.9.1 Detailed Description

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

3.9.2 Constructor & Destructor Documentation

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

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.9.2.2 IETF_CSTAp1 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.10 IETF_CSTAp2 Class Reference

Inherits Com::Objsys::Csta::Phase2::GenericCSTAp2.

Inherited by Alcatel4400.

Public Member Functions

- IETF_CSTAp2 (PBXSession sessionObject)
- IETF_CSTAp2 (string pbxSystem, int port)

3.10.1 Detailed Description

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

3.10.2 Constructor & Destructor Documentation

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

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.10.2.2 IETF_CSTAp2 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.11 IETF_CSTAp3 Class Reference

Inherits Com::Objsys::Csta::Phase3::GenericCSTAp3.

Inherited by PanasonicKXTDE, and SiemensCap.

Public Member Functions

- IETF_CSTAp3 (PBXSession sessionObject)
- IETF_CSTAp3 (string pbxSystem, int port)

3.11.1 Detailed Description

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

3.11.2 Constructor & Destructor Documentation

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

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.11.2.2 IETF_CSTAp3 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.12 LicenseException Class Reference

3.12.1 Detailed Description

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

3.13 PanasonicKXTDA Class Reference

Inherits Com::Objsys::Csta::Devices::PanasonicKXTDE.

Public Member Functions

- PanasonicKXTDA (PBXSession sessionObject)
- PanasonicKXTDA (string pbxSystem, int port)

3.13.1 Detailed Description

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

3.13.2 Constructor & Destructor Documentation

3.13.2.1 PanasonicKXTDA (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.13.2.2 PanasonicKXTDA (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.14 PanasonicKXTDE Class Reference

Inherits Com::Objsys::Csta::Phase3::IETF_CSTAp3.

Inherited by PanasonicKXTDA, and PanasonicNCP.

Public Types

- enum DeviceDataTypes

Public Member Functions

- CSTAResponseInfo AcquireControlRight (string targetDevice)
- CSTAResponseInfo ClearMessageWaiting (string originatingDevice, string targetDevice)
- CSTAResponseInfo GetDeviceData (string device, DeviceDataTypes eDataType)
- CSTAResponseInfo GetGroupMembers (string groupDevice)
- override CSTAResponseInfo GetSFDevices ()
- PanasonicKXTDE (PBXSession sessionObject)
- PanasonicKXTDE (string pbxSystem, int port)
- CSTAResponseInfo PDFStart (string targetDevice)
- CSTAResponseInfo PDFStop (string targetDevice)
- CSTAResponseInfo ReleaseControlRight (string targetDevice)
- CSTAResponseInfo ResetDisplay (string targetDevice)
- CSTAResponseInfo SendKmeMessage (Asn1BerEncodeBuffer encodeBuffer)
- CSTAResponseInfo SetMessageWaiting (string originatingDevice, string targetDevice)

3.14.1 Detailed Description

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

3.14.2 Member Enumeration Documentation

3.14.2.1 enum DeviceDataTypes

Provides symbolic names for the types of device information tha can be obtained from a Panasonic PBX.

3.14.3 Constructor & Destructor Documentation

3.14.3.1 PanasonicKXTDE (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.14.3.2 PanasonicKXTDE (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.14.4 Member Function Documentation

3.14.4.1 CSTAResponseInfo AcquireControlRight (string *targetDevice*)

Acquires the right to control a telephony device. This method will cause a "PDF (Physical Device Feature) Start" Escape message to be sent to the PBX.

Parameters

targetDevice The target telephony device.

Returns

A CSTAResponseInfo object.

3.14.4.2 CSTAResponseInfo ClearMessageWaiting (string *originatingDevice*, string *targetDevice*)

Turns off a device's message waiting indicator.

Parameters

originatingDevice The device that originated the call back request.

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

Returns

A CSTAResponseInfo object.

3.14.4.3 CSTAResponseInfo GetDeviceData (string *device*, DeviceDataTypes *eDataType*)

Gets information about a device.

Parameters

device The device (e.g., "101") about which the information is desired.

eDataType The type of data requested (must be from the DeviceDataTypes enum in this class).

Returns

3.14.4.4 CSTAResponseInfo GetGroupMembers (string *groupDevice*)

Gets the members associated with a group device (for example, the extensions associated with an incoming call distribution group device).

Parameters

groupDevice The group device designation (e.g, "601").

Returns

A CSTAResponseInfo object.

3.14.4.5 override CSTAResponseInfo GetSFDevices () `[virtual]`

Returns a list of station (i.e., telephone) devices known to the PBX by sending a Get Switching Function Devices message that specifies just station devices.

Returns

A CSTAResponseInfo object.

Reimplemented from GenericCSTAp3.

3.14.4.6 CSTAResponseInfo PDFStart (string *targetDevice*)

Sends a "PDF (Physical Device Feature) Start" Escape message to the PBX. This method is a convenience method that does the same thing as the AcquireControlRight() method.

Parameters

targetDevice The target telephony device.

Returns

A CSTAResponseInfo object.

3.14.4.7 CSTAResponseInfo PDFStop (string *targetDevice*)

Sends a "PDF (Physical Device Feature) Stop" Escape message to the PBX. This method is a convenience method that does the same thing as the ReleaseControlRight() method.

Parameters

targetDevice The target telephony device.

Returns

A CSTAResponseInfo object.

3.14.4.8 CSTAResponseInfo ReleaseControlRight (string *targetDevice*)

Releases the right to control a telephony device. This method will cause a "PDF (Physical Device Feature) Stop" Escape message to be sent to the PBX.

Parameters

targetDevice The target telephony device.

Returns

A CSTAResponseInfo object.

3.14.4.9 CSTAResponseInfo ResetDisplay (string *targetDevice*)

Resets a telephony device's display

Parameters

targetDevice The device which is to be reset.

Returns

A CSTAResponseInfo object.

3.14.4.10 CSTAResponseInfo SendKmeMessage (Asn1BerEncodeBuffer *encodeBuffer*)

Sends a Panasonic-specific KME message to the PBX.

Parameters

encodeBuffer A BER encode buffer instance containing a completely encoded KME message. The message must include the KMESpecificPrivateData segment, the EscapeArgument, and the ROSE header.

Returns

3.14.4.11 CSTAResponseInfo SetMessageWaiting (string *originatingDevice*, string *targetDevice*)

Turns on a device's message waiting indicator.

Parameters

originatingDevice The device that originated the call back request.

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

Returns

A CSTAResponseInfo object.

3.15 PanasonicNCP Class Reference

Inherits Com::Objsys::Csta::Devices::PanasonicKXTDE.

Public Member Functions

- PanasonicNCP (PBXSession sessionObject)
- PanasonicNCP (string pbxSystem, int port)

3.15.1 Detailed Description

Implements CSTA phase 3 operations for the Panasonic NCP PBX device.

3.15.2 Constructor & Destructor Documentation

3.15.2.1 PanasonicNCP (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.15.2.2 PanasonicNCP (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.16 PBXSession Class Reference

Public Member Functions

- delegate void AsyncCallback (PBXSession sessionObject, byte[ ] asyncData)
- void Close (CSTAContext threadContext)
- delegate void ConnectionCallback (PBXSession sessionObject)
- void Open (CSTAContext threadContext)
- PBXSession (string pbxSystem, int port)
- SocketState SendACSEMessage (byte[ ] message, int messageLength, Constants.ACSEMessageTypes messageType, CSTAContext threadContext)
- void SendMessage (string messageType, byte[ ] message, int messageLength, CSTAContext threadContext)
- void SendMessage (byte[ ] message, int messageLength, CSTAContext threadContext)
- void WaitForROSEResponse (CSTAContext threadContext)

Properties

- AsyncCallback ClientCallback `[get, set]`
- ConnectionCallback ConnectionLostCallback `[get, set]`
- bool DebugClientCallback `[get, set]`
- bool DebugMode `[get, set]`
- int MaxReceiveTimeout `[get, set]`
- string PBXSystem `[get]`
- int Port `[get]`

3.16.1 Detailed Description

This class manages communication with a PBX. One instance of this class should be created for each PBX with which a CSTADLL client application needs to exchange CSTA messages.

The CSTA worker classes (e.g., Alcatel4400, PanasonicNCP) hold a reference to a PBXSession object. If the constructor for the worker class that takes a PBX identification and a PBX port is used, a PBXSession object is created. Alternatively, the client application can create a PBXSession instance and pass a reference to the instance to the other worker class constructor signature.

Only one PBXSession instance for a PBX/port combination should be created. The behavior is undefined if multiple PBXSession instances are created for the same PBX and port.

3.16.2 Constructor & Destructor Documentation

3.16.2.1 PBXSession (string *pbxSystem*, int *port*)

Constructs a PBXSession object.

Parameters

pbxSystem The name or IP address of the PBX system.

port The port on the PBX system to which the client is connecting.

3.16.3 Member Function Documentation

3.16.3.1 delegate void AsyncCallback (PBXSession *sessionObject*, byte[ ] *asyncData*)

Declaration of a callback function to be invoked when an asynchronous message is received, such as from a monitor session.

Parameters

sessionObject The session object for the PBX that generated the asynchronous message.

asyncData The data received asynchronously from the PBX.

3.16.3.2 void Close (CSTAContext *threadContext*)

Terminates the session to the PBX. This method can be used to terminate sessions with PBX devices that don't accept ACSE release association requests.

Parameters

threadContext The context object for the caling thread.

3.16.3.3 delegate void ConnectionCallback (PBXSession *sessionObject*)

Declaration of a callback function to be invoked if the connection to the PBX is lost.

Parameters

sessionObject The session object for the PBX whose connection was lost.

3.16.3.4 void Open (CSTAContext *threadContext*)

This method can be used to establish communication with a PBX device before any messages are actually sent to the device.

Parameters

threadContext The thread context object.

3.16.3.5 SocketState SendACSEMessage (byte[ ] *message*, int *messageLength*, Constants.ACSEMessageTypes *messageType*, CSTAContext *threadContext*)

This method sends an ACSE message (either Make Association or Release Association) to the PBX and receives the response. This operation is done synchronously. If the Make Association needs to be done (usually it does), it must be done before any threads for sending and receiving CSTA messages are started.

This method is only intended to be used by client code that encodes its own ACSEMakeAssociation or ACSEReleaseAssociation message. Most clients can probably use the MakeACSEAssociation() and ReleaseACSEAssociation() methods that are in each phase's helper classes.

Parameters

message An encoded ACSE Make Association or Release Association message.

messageLength The length of the encoded message.

messageType A constant telling whether the message is an ACSE Make Association or an ACSE Release Association.

threadContext The thread context object.

Returns

A populated SocketState instance.

3.16.3.6 void SendMessage (string *messageType*, byte[ ] *message*, int *messageLength*, CSTAContext *threadContext*)

This method sends a message to the PBX using TCP/IP.

Parameters

messageType A string token to help identify the message in the CSTADLL log file.

message Byte array containing the encoded message to send.

messageLength The length of the encoded message.

threadContext The thread context object.

3.16.3.7 void SendMessage (byte[ ] *message*, int *messageLength*, CSTAContext *threadContext*)

This method sends a message to the PBX using TCP/IP.

Parameters

message Byte array containing the encoded message to send.

messageLength The length of the encoded message.

threadContext The thread context object.

3.16.3.8 void WaitForROSEResponse (CSTAContext *threadContext*)

This method waits for a response to a CSTA message sent with a ROSE header.

Parameters

threadContext The CSTAContext object associated with the calling thread.

3.16.4 Property Documentation

3.16.4.1 AsyncCallback ClientCallback `[get, set]`

Holds a reference to an asynchronous callback function. This function will be invoked if data is received asynchronously from the PBX, such as from a monitor operation.

3.16.4.2 ConnectionCallback ConnectionLostCallback `[get, set]`

Holds a reference to an asynchronous callback function. This function will be invoked if the connection to the PBX is lost.

3.16.4.3 bool DebugClientCallback [get, set]

Enables easier debugging of asynchronous callback methods. If set to true, the read for the next message from the PBX won't occur until after the client callback method returns. Normally the read for the next message occurs before the client callback method is invoked. The default value for this property is false.

3.16.4.4 bool DebugMode [get, set]

Enables behavior that facilitates debugging of the CSTADLL software. This property is most likely useful only to Objective Systems staff.

3.16.4.5 int MaxReceiveTimeout [get, set]

Specifies the amount of time, in milliseconds, to wait for a respone to arrive from the PBX. The default value is 5,000 milliseconds (5 seconds).

3.16.4.6 string PBXSystem [get]

The TCIP/IP address or well-known name of the PBX.

3.16.4.7 int Port [get]

The port where the PBX listens for CSTA messages.

3.17 PBXSessionException Class Reference

3.17.1 Detailed Description

Defines an exception that occurs while communicating with a PBX.

3.18 PBXSessionHelper Class Reference

Properties

- static bool LoggingEnabled `[get, set]`
- static string LoggingFolder `[get, set]`

3.18.1 Detailed Description

This class holds static properties that affect all PBX sessions.

3.18.2 Property Documentation

3.18.2.1 bool LoggingEnabled `[static, get, set]`

Determines whether logging of traffic between the client and the PBX will be done. This is a static property that affects all connections to all PBXes.

3.18.2.2 string LoggingFolder `[static, get, set]`

Specifies a folder to receive the log file. If not specified, the log file will go into whatever folder the calling .exe resides in.

3.19 Phase1Opcodes Class Reference

Public Types

- enum Opcodes

3.19.1 Detailed Description

This class contains a public enum that contains symbolic names for the opcodes that define CSTA phase 1 operations.

3.19.2 Member Enumeration Documentation

3.19.2.1 enum Opcodes

Contains symbolic names for the opcodes that define CSTA phase 1 operations.

3.20 Phase2Opcodes Class Reference

Public Types

- enum Opcodes

3.20.1 Detailed Description

This class contains a public enum that contains symbolic names for the opcodes that define CSTA phase 2 operations.

3.20.2 Member Enumeration Documentation

3.20.2.1 enum Opcodes

Contains symbolic names for the opcodes that define CSTA phase 2 operations.

3.21 Phase3Opcodes Class Reference

Public Types

- enum Opcodes

3.21.1 Detailed Description

This class contains a public enum that contains symbolic names for the opcodes that define CSTA phase 3 operations.

3.21.2 Member Enumeration Documentation

3.21.2.1 enum Opcodes

Contains symbolic names for the opcodes that define CSTA phase 3 operations.

3.22 PhilipsSopho Class Reference

Inherits Com::Objsys::Csta::Phase1::IETF_CSTAp1.

Public Member Functions

- CSTAResponseInfo MakeACSEAssociation (bool testMode)
- override CSTAResponseInfo MakeACSEAssociation ()
- PhilipsSopho (PBXSession sessionObject)
- PhilipsSopho (string pbxSystem, int port)

3.22.1 Detailed Description

Implements CSTA phase 1 operations for the Philips Sopho PBX device.

3.22.2 Constructor & Destructor Documentation

3.22.2.1 PhilipsSopho (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.22.2.2 PhilipsSopho (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.22.3 Member Function Documentation

3.22.3.1 CSTAResponseInfo MakeACSEAssociation (bool *testMode*)

Establish an ACSE association with the PBX.

Parameters

testMode True to send a test mode message, false to send a licensed message.

Returns

A CSTAResponseInfo object.

3.22.3.2 override CSTAResponseInfo MakeACSEAssociation () [virtual]

Establish a licensed (i.e., not test mode) ACSE association with the PBX.

Returns

A CSTAResponseInfo object.

Reimplemented from GenericCSTAp1.

3.23 ROSEParseInfo Class Reference

3.23.1 Detailed Description

Contains information about a received CSTA message obtained by decoding the message's ROSE header.

3.24 SiemensCap Class Reference

Inherits Com::Objsys::Csta::Phase3::IETF_CSTAp3.

Inherited by SiemensHipath4000.

Public Member Functions

- SiemensCap (PBXSession sessionObject)
- SiemensCap (string pbxSystem, int port)

3.24.1 Detailed Description

Implements CSTA phase 3 operations for the Siemens CAP PBX device.

3.24.2 Constructor & Destructor Documentation

3.24.2.1 SiemensCap (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.24.2.2 SiemensCap (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.25 SiemensHicom300 Class Reference

Inherits Com::Objsys::Csta::Phase1::GenericCSTAp1.

Public Member Functions

- SiemensHicom300 (PBXSession sessionObject)
- SiemensHicom300 (string pbxSystem, int port)

3.25.1 Detailed Description

Implements CSTA phase 1 operations for the Siemens Hicom 300 PBX device.

3.25.2 Constructor & Destructor Documentation

3.25.2.1 SiemensHicom300 (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.25.2.2 SiemensHicom300 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.26 SiemensHipath3000p2 Class Reference

Inherits Com::Objsys::Csta::Phase2::GenericCSTAp2.

Public Member Functions

- SiemensHipath3000p2 (PBXSession sessionObject)
- SiemensHipath3000p2 (string pbxSystem, int port)

3.26.1 Detailed Description

Implements CSTA phase 2 operations for the Siemens Hipath 3000 PBX device.

3.26.2 Constructor & Destructor Documentation

3.26.2.1 SiemensHipath3000p2 (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.26.2.2 SiemensHipath3000p2 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.27 SiemensHipath3000p3 Class Reference

Inherits Com::Objsys::Csta::Phase3::GenericCSTAp3.

Inherited by UnifyOpenscapeX5.

Public Member Functions

- virtual CSTAResponseInfo MakeACSEAssociation (string userName, string password)
- SiemensHipath3000p3 (PBXSession sessionObject)
- SiemensHipath3000p3 (string pbxSystem, int port)

3.27.1 Detailed Description

Implements CSTA phase 3 operations for the Siemens Hipath 3000 PBX device.

3.27.2 Constructor & Destructor Documentation

3.27.2.1 SiemensHipath3000p3 (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.27.2.2 SiemensHipath3000p3 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.27.3 Member Function Documentation

3.27.3.1 virtual CSTAResponseInfo MakeACSEAssociation (string *userName*, string *password*) [virtual]

Establish an ACSE association with the PBX.

Parameters

userName The user name to send to the PBX.

password The password to send to the PBX.

Returns

A CSTAResponseInfo object.

3.28 SiemensHipath4000 Class Reference

Inherits Com::Objsys::Csta::Devices::SiemensCap.

Public Member Functions

- SiemensHipath4000 (PBXSession sessionObject)
- SiemensHipath4000 (string pbxSystem, int port)

3.28.1 Detailed Description

Implements CSTA phase 3 operations for the Siemens Hipath 4000 PBX device.

3.28.2 Constructor & Destructor Documentation

3.28.2.1 SiemensHipath4000 (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.28.2.2 SiemensHipath4000 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.29 SiemensRealitis Class Reference

Inherits Com::Objsys::Csta::Phase1::IETF_CSTAp1.

Public Member Functions

- SiemensRealitis (PBXSession sessionObject)
- SiemensRealitis (string pbxSystem, int port)

3.29.1 Detailed Description

Implements CSTA phase 1 operations for the Siemens Realitis PBX device.

3.29.2 Constructor & Destructor Documentation

3.29.2.1 SiemensRealitis (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.29.2.2 SiemensRealitis (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.30 SingleStepTransferInfo Class Reference

Properties

- string TransferFromDevice `[get, set]`
- string TransferringCallID `[get, set]`
- string TransferToDevice `[get, set]`

3.30.1 Detailed Description

Contains information needed to complete a phase 3 single step transfer request.

3.30.2 Property Documentation

3.30.2.1 string TransferFromDevice `[get, set]`

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

3.30.2.2 string TransferringCallID `[get, set]`

The call id number associated with the device from which the call is being transferred.

3.30.2.3 string TransferToDevice `[get, set]`

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

3.31 SocketState Class Reference

Properties

- byte[ ] AckBuffer `[get, set]`
- byte[ ] ReadBuffer `[get, set]`
- List< byte[ ]> ReadBuffers `[get, set]`
- int TotalLength `[get, set]`

3.31.1 Detailed Description

This class contains the response received from the PBX and state information about the exchange with the PBX that is used internally by CSTADLL.

3.31.2 Property Documentation

3.31.2.1 byte [ ] AckBuffer `[get, set]`

Contains the first response from the PBX for situations where the PBX sends multiple response messages (e.g., Get Switching Function Devices). The data messages that are sent after this ack will be in ReadBuffers.

3.31.2.2 byte [ ] ReadBuffer `[get, set]`

Contains the bytes most recently read from the socket. This buffer will be filled in bit by bit as the message is read.

3.31.2.3 List<byte[ ]> ReadBuffers `[get, set]`

Contains multiple collections of bytes read from the socket. This array is used for situations where a response to a message comes in multiple segments (e.g., Get Switching Function Devices). For these situations the immediate response will be in AckBuffer.

3.31.2.4 int TotalLength `[get, set]`

The total length of a complete message received from the PBX. This is also used as an offset into the read buffer so we can build the message as it's received.

3.32 TadiranCoral Class Reference

Inherits Com::Objsys::Csta::Phase1::IETF_CSTAp1.

Public Member Functions

- TadiranCoral (PBXSession sessionObject)
- TadiranCoral (string pbxSystem, int port)

3.32.1 Detailed Description

Implements CSTA phase 1 operations for the Tadiran Coral PBX device.

3.32.2 Constructor & Destructor Documentation

3.32.2.1 TadiranCoral (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.32.2.2 TadiranCoral (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

3.33 TransferCallInfo Class Reference

Properties

- string ActiveDeviceCallID `[get, set]`
- string ActiveDeviceNumber `[get, set]`
- string ConnectedDevice `[get, set]`
- string HeldDevice `[get, set]`
- string HeldDeviceCallID `[get, set]`
- string HeldDeviceNumber `[get, set]`

3.33.1 Detailed Description

Contains information needed to complete a phase 3 transfer call request.

3.33.2 Property Documentation

3.33.2.1 string ActiveDeviceCallID `[get, set]`

The call id associated with the device to which the call is being transferred.

3.33.2.2 string ActiveDeviceNumber `[get, set]`

The phone number to which the call is being transferred. This number is not necessarily the same as the value for ConnectedDevice.

3.33.2.3 string ConnectedDevice `[get, set]`

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

3.33.2.4 string HeldDevice `[get, set]`

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

3.33.2.5 string HeldDeviceCallID `[get, set]`

The call id associated with the device from which the call is being transferred.

3.33.2.6 string HeldDeviceNumber `[get, set]`

The phone number from which the call is being transferred. This number is not necessarily the same as the value for HeldDevice.

3.34 UnifyOpenscapeX5 Class Reference

Inherits Com::Objsys::Csta::Devices::SiemensHipath3000p3.

Public Member Functions

- UnifyOpenscapeX5 (PBXSession sessionObject)
- UnifyOpenscapeX5 (string pbxSystem, int port)

3.34.1 Detailed Description

Implements CSTA phase 3 operations for the Unify Openscape Business X5 PBX device.

3.34.2 Constructor & Destructor Documentation

3.34.2.1 UnifyOpenscapeX5 (string *pbxSystem*, int *port*)

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

Parameters

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

port Port on which the PBX listens for CSTA messages.

3.34.2.2 UnifyOpenscapeX5 (PBXSession *sessionObject*)

Constructs an instance associated with the given PBXSession object.

Parameters

sessionObject A PBXSession object.

Index

AckBuffer
    Com::Objsys::Csta::Common::SocketState, 69
AcquireControlRight
    Com::Objsys::Csta::Devices::PanasonicKXTDE, 46
ActiveDeviceCallID
    Com::Objsys::Csta::Phase3::TransferCallInfo, 71
ActiveDeviceNumber
    Com::Objsys::Csta::Phase3::TransferCallInfo, 71
Alcatel4400
    Com::Objsys::Csta::Devices::Alcatel4400, 9
AlcatelOXE
    Com::Objsys::Csta::Devices::AlcatelOXE, 11
AlcatelOXO
    Com::Objsys::Csta::Devices::AlcatelOXO, 12
AnswerCall
    Com::Objsys::Csta::Phase1::GenericCSTAp1, 16
    Com::Objsys::Csta::Phase2::GenericCSTAp2, 23
    Com::Objsys::Csta::Phase3::GenericCSTAp3, 30
AsyncCallback
    Com::Objsys::Csta::Common::PBXSession, 51

ClearConnection
    Com::Objsys::Csta::Phase1::GenericCSTAp1, 17
    Com::Objsys::Csta::Phase2::GenericCSTAp2, 24
    Com::Objsys::Csta::Phase3::GenericCSTAp3, 31
ClearDoNotDisturb
    Com::Objsys::Csta::Phase1::GenericCSTAp1, 17
    Com::Objsys::Csta::Phase2::GenericCSTAp2, 24
    Com::Objsys::Csta::Phase3::GenericCSTAp3, 31
ClearMessageWaiting
    Com::Objsys::Csta::Devices::PanasonicKXTDE, 46
    Com::Objsys::Csta::Phase1::GenericCSTAp1, 17
    Com::Objsys::Csta::Phase2::GenericCSTAp2, 24
    Com::Objsys::Csta::Phase3::GenericCSTAp3, 31
ClientCallback
    Com::Objsys::Csta::Common::PBXSession, 52
Close
    Com::Objsys::Csta::Common::PBXSession, 51
Com.Objsys.Csta.Common, 3
Com.Objsys.Csta.Devices, 4
Com.Objsys.Csta.Phase1, 5
Com.Objsys.Csta.Phase2, 6
Com.Objsys.Csta.Phase3, 7
Com::Objsys::Csta::Common::CSTAContext, 13
    ResponseFromPBX, 13
    ResponsesFromPBX, 13
Com::Objsys::Csta::Common::CSTAResponseInfo, 14
    ResponseFromPBX, 14
    ResponsesFromPBX, 14
    StatusCode, 14
    StatusMessage, 14
Com::Objsys::Csta::Common::LicenseException, 43
Com::Objsys::Csta::Common::PBXSession, 50
    AsyncCallback, 51
    ClientCallback, 52
    Close, 51
    ConnectionCallback, 51
    ConnectionLostCallback, 52
    DebugClientCallback, 52
    DebugMode, 53
    MaxReceiveTimeout, 53
    Open, 51
    PBXSession, 50
    PBXSystem, 53
    Port, 53
    SendACSEMessage, 51
    SendMessage, 52
    WaitForROSEResponse, 52
Com::Objsys::Csta::Common::PBXSessionException, 54
Com::Objsys::Csta::Common::PBXSessionHelper, 55
    LoggingEnabled, 55
    LoggingFolder, 55
Com::Objsys::Csta::Common::ROSEParseInfo, 61
Com::Objsys::Csta::Common::SocketState, 69
    AckBuffer, 69
    ReadBuffer, 69
    ReadBuffers, 69
    TotalLength, 69
Com::Objsys::Csta::Devices::Alcatel4400, 9
    Alcatel4400, 9
    MakeACSEAssociation, 10
Com::Objsys::Csta::Devices::AlcatelOXE, 11
    AlcatelOXE, 11
Com::Objsys::Csta::Devices::AlcatelOXO, 12
    AlcatelOXO, 12
    MakeACSEAssociation, 12
Com::Objsys::Csta::Devices::PanasonicKXTDA, 44
    PanasonicKXTDA, 44
Com::Objsys::Csta::Devices::PanasonicKXTDE, 45
    AcquireControlRight, 46

ClearMessageWaiting, 46
DeviceDataTypes, 45
GetDeviceData, 46
GetGroupMembers, 46
GetSFDevices, 47
PanasonicKXTDE, 45
PDFStart, 47
PDFStop, 47
ReleaseControlRight, 47
ResetDisplay, 48
SendKmeMessage, 48
SetMessageWaiting, 48
Com::Objsys::Csta::Devices::PanasonicNCP, 49
PanasonicNCP, 49
Com::Objsys::Csta::Devices::PhilipsSopho, 59
MakeACSEAssociation, 59
PhilipsSopho, 59
Com::Objsys::Csta::Devices::SiemensCap, 62
SiemensCap, 62
Com::Objsys::Csta::Devices::SiemensHicom300, 63
SiemensHicom300, 63
Com::Objsys::Csta::Devices::SiemensHipath3000p2, 64
SiemensHipath3000p2, 64
Com::Objsys::Csta::Devices::SiemensHipath3000p3, 65
MakeACSEAssociation, 65
SiemensHipath3000p3, 65
Com::Objsys::Csta::Devices::SiemensHipath4000, 66
SiemensHipath4000, 66
Com::Objsys::Csta::Devices::SiemensRealitis, 67
SiemensRealitis, 67
Com::Objsys::Csta::Devices::TadiranCoral, 70
TadiranCoral, 70
Com::Objsys::Csta::Devices::UnifyOpenscapeX5, 72
UnifyOpenscapeX5, 72
Com::Objsys::Csta::Phase1::GenericCSTAp1, 15
AnswerCall, 16
ClearConnection, 17
ClearDoNotDisturb, 17
ClearMessageWaiting, 17
ConsultationCall, 17
DivertCall, 17
EncodeROSERequestHeader, 18
GenericCSTAp1, 16
MakeACSEAssociation, 18
MakeCall, 18
MonitorStart, 18
MonitorStop, 19
QueryDevice, 19
ReleaseACSEAssociation, 20
RetrieveCall, 20
SessionObject, 21
SetDoNotDisturb, 20
SetMessageWaiting, 20
SnapshotDevice, 20
ThreadContext, 21
TransferCall, 21
Com::Objsys::Csta::Phase1::IETF_CSTAp1, 40
IETF_CSTAp1, 40
Com::Objsys::Csta::Phase1::Phase1Opcodes, 56
Opcodes, 56
Com::Objsys::Csta::Phase2::GenericCSTAp2, 22
AnswerCall, 23
ClearConnection, 24
ClearDoNotDisturb, 24
ClearMessageWaiting, 24
ConsultationCall, 24
DivertCall, 24
EncodeROSERequestHeader, 25
GenericCSTAp2, 23
MakeACSEAssociation, 25
MakeCall, 25
MonitorStart, 25
MonitorStop, 26
QueryDevice, 26
ReleaseACSEAssociation, 27
RetrieveCall, 27
SessionObject, 28
SetDoNotDisturb, 27
SetMessageWaiting, 27
SnapshotDevice, 27
ThreadContext, 28
TransferCall, 28
Com::Objsys::Csta::Phase2::IETF_CSTAp2, 41
IETF_CSTAp2, 41
Com::Objsys::Csta::Phase2::Phase2Opcodes, 57
Opcodes, 57
Com::Objsys::Csta::Phase3::GenericCSTAp3, 29
AnswerCall, 30
ClearConnection, 31
ClearDoNotDisturb, 31
ClearMessageWaiting, 31
ConsultationCall, 31
EncodeROSERequestHeader, 32
GenericCSTAp3, 30
GetAgentState, 32
GetDoNotDisturb, 32
GetSFDevices, 32, 33
HoldCall, 33
MakeACSEAssociation, 33
MakeCall, 33
MonitorStart, 33
MonitorStop, 34
ReleaseACSEAssociation, 34
RetrieveCall, 35
RingDevice, 35
SendData, 35
SessionObject, 39
SetDisplay, 35

SetDoNotDisturb, 36
SetMessageWaiting, 36
SingleStepTransfer, 36
SnapshotCall, 37
SnapshotDevice, 37
StartDataPath, 37
StopDataPath, 37
StopRing, 38
ThreadContext, 39
TransferCall, 38
Com::Objsys::Csta::Phase3::IETF_CSTAp3, 42
IETF_CSTAp3, 42
Com::Objsys::Csta::Phase3::Phase3Opcodes, 58
Opcodes, 58
Com::Objsys::Csta::Phase3::SingleStepTransferInfo, 68
TransferFromDevice, 68
TransferringCallID, 68
TransferToDevice, 68
Com::Objsys::Csta::Phase3::TransferCallInfo, 71
ActiveDeviceCallID, 71
ActiveDeviceNumber, 71
ConnectedDevice, 71
HeldDevice, 71
HeldDeviceCallID, 71
HeldDeviceNumber, 71
ConnectedDevice
Com::Objsys::Csta::Phase3::TransferCallInfo, 71
ConnectionCallback
Com::Objsys::Csta::Common::PBXSession, 51
ConnectionLostCallback
Com::Objsys::Csta::Common::PBXSession, 52
ConsultationCall
Com::Objsys::Csta::Phase1::GenericCSTAp1, 17
Com::Objsys::Csta::Phase2::GenericCSTAp2, 24
Com::Objsys::Csta::Phase3::GenericCSTAp3, 31

DebugClientCallback
Com::Objsys::Csta::Common::PBXSession, 52
DebugMode
Com::Objsys::Csta::Common::PBXSession, 53
DeviceDataTypes
Com::Objsys::Csta::Devices::PanasonicKXTDE, 45
DivertCall
Com::Objsys::Csta::Phase1::GenericCSTAp1, 17
Com::Objsys::Csta::Phase2::GenericCSTAp2, 24

EncodeROSERequestHeader
Com::Objsys::Csta::Phase1::GenericCSTAp1, 18
Com::Objsys::Csta::Phase2::GenericCSTAp2, 25
Com::Objsys::Csta::Phase3::GenericCSTAp3, 32

GenericCSTAp1
Com::Objsys::Csta::Phase1::GenericCSTAp1, 16
GenericCSTAp2
Com::Objsys::Csta::Phase2::GenericCSTAp2, 23
GenericCSTAp3
Com::Objsys::Csta::Phase3::GenericCSTAp3, 30
GetAgentState
Com::Objsys::Csta::Phase3::GenericCSTAp3, 32
GetDeviceData
Com::Objsys::Csta::Devices::PanasonicKXTDE, 46
GetDoNotDisturb
Com::Objsys::Csta::Phase3::GenericCSTAp3, 32
GetGroupMembers
Com::Objsys::Csta::Devices::PanasonicKXTDE, 46
GetSFDevices
Com::Objsys::Csta::Devices::PanasonicKXTDE, 47
Com::Objsys::Csta::Phase3::GenericCSTAp3, 32, 33

HeldDevice
Com::Objsys::Csta::Phase3::TransferCallInfo, 71
HeldDeviceCallID
Com::Objsys::Csta::Phase3::TransferCallInfo, 71
HeldDeviceNumber
Com::Objsys::Csta::Phase3::TransferCallInfo, 71
HoldCall
Com::Objsys::Csta::Phase3::GenericCSTAp3, 33

IETF_CSTAp1
Com::Objsys::Csta::Phase1::IETF_CSTAp1, 40
IETF_CSTAp2
Com::Objsys::Csta::Phase2::IETF_CSTAp2, 41
IETF_CSTAp3
Com::Objsys::Csta::Phase3::IETF_CSTAp3, 42

LoggingEnabled
Com::Objsys::Csta::Common::PBXSessionHelper, 55
LoggingFolder
Com::Objsys::Csta::Common::PBXSessionHelper, 55

MakeACSEAssociation
Com::Objsys::Csta::Devices::Alcatel4400, 10
Com::Objsys::Csta::Devices::AlcatelOXO, 12
Com::Objsys::Csta::Devices::PhilipsSopho, 59
Com::Objsys::Csta::Devices::SiemensHipath3000p3, 65
Com::Objsys::Csta::Phase1::GenericCSTAp1, 18
Com::Objsys::Csta::Phase2::GenericCSTAp2, 25
Com::Objsys::Csta::Phase3::GenericCSTAp3, 33
MakeCall
Com::Objsys::Csta::Phase1::GenericCSTAp1, 18
Com::Objsys::Csta::Phase2::GenericCSTAp2, 25
Com::Objsys::Csta::Phase3::GenericCSTAp3, 33
MaxReceiveTimeout
Com::Objsys::Csta::Common::PBXSession, 53
MonitorStart
Com::Objsys::Csta::Phase1::GenericCSTAp1, 18

Com::Objsys::Csta::Phase2::GenericCSTAp2, 25
Com::Objsys::Csta::Phase3::GenericCSTAp3, 33
MonitorStop
Com::Objsys::Csta::Phase1::GenericCSTAp1, 19
Com::Objsys::Csta::Phase2::GenericCSTAp2, 26
Com::Objsys::Csta::Phase3::GenericCSTAp3, 34

Opcodes
Com::Objsys::Csta::Phase1::Phase1Opcodes, 56
Com::Objsys::Csta::Phase2::Phase2Opcodes, 57
Com::Objsys::Csta::Phase3::Phase3Opcodes, 58
Open
Com::Objsys::Csta::Common::PBXSession, 51

PanasonicKXTDA
Com::Objsys::Csta::Devices::PanasonicKXTDA, 44
PanasonicKXTDE
Com::Objsys::Csta::Devices::PanasonicKXTDE, 45
PanasonicNCP
Com::Objsys::Csta::Devices::PanasonicNCP, 49
PBXSession
Com::Objsys::Csta::Common::PBXSession, 50
PBXSystem
Com::Objsys::Csta::Common::PBXSession, 53
PDFStart
Com::Objsys::Csta::Devices::PanasonicKXTDE, 47
PDFStop
Com::Objsys::Csta::Devices::PanasonicKXTDE, 47
PhilipsSopho
Com::Objsys::Csta::Devices::PhilipsSopho, 59
Port
Com::Objsys::Csta::Common::PBXSession, 53

QueryDevice
Com::Objsys::Csta::Phase1::GenericCSTAp1, 19
Com::Objsys::Csta::Phase2::GenericCSTAp2, 26

ReadBuffer
Com::Objsys::Csta::Common::SocketState, 69
ReadBuffers
Com::Objsys::Csta::Common::SocketState, 69
ReleaseACSEAssociation
Com::Objsys::Csta::Phase1::GenericCSTAp1, 20
Com::Objsys::Csta::Phase2::GenericCSTAp2, 27
Com::Objsys::Csta::Phase3::GenericCSTAp3, 34
ReleaseControlRight
Com::Objsys::Csta::Devices::PanasonicKXTDE, 47
ResetDisplay
Com::Objsys::Csta::Devices::PanasonicKXTDE, 48
ResponseFromPBX
Com::Objsys::Csta::Common::CSTAContext, 13
Com::Objsys::Csta::Common::CSTAResponseInfo, 14
ResponsesFromPBX
Com::Objsys::Csta::Common::CSTAContext, 13
Com::Objsys::Csta::Common::CSTAResponseInfo, 14
RetrieveCall
Com::Objsys::Csta::Phase1::GenericCSTAp1, 20
Com::Objsys::Csta::Phase2::GenericCSTAp2, 27
Com::Objsys::Csta::Phase3::GenericCSTAp3, 35
RingDevice
Com::Objsys::Csta::Phase3::GenericCSTAp3, 35

SendACSEMessage
Com::Objsys::Csta::Common::PBXSession, 51
SendData
Com::Objsys::Csta::Phase3::GenericCSTAp3, 35
SendKmeMessage
Com::Objsys::Csta::Devices::PanasonicKXTDE, 48
SendMessage
Com::Objsys::Csta::Common::PBXSession, 52
SessionObject
Com::Objsys::Csta::Phase1::GenericCSTAp1, 21
Com::Objsys::Csta::Phase2::GenericCSTAp2, 28
Com::Objsys::Csta::Phase3::GenericCSTAp3, 39
SetDisplay
Com::Objsys::Csta::Phase3::GenericCSTAp3, 35
SetDoNotDisturb
Com::Objsys::Csta::Phase1::GenericCSTAp1, 20
Com::Objsys::Csta::Phase2::GenericCSTAp2, 27
Com::Objsys::Csta::Phase3::GenericCSTAp3, 36
SetMessageWaiting
Com::Objsys::Csta::Devices::PanasonicKXTDE, 48
Com::Objsys::Csta::Phase1::GenericCSTAp1, 20
Com::Objsys::Csta::Phase2::GenericCSTAp2, 27
Com::Objsys::Csta::Phase3::GenericCSTAp3, 36
SiemensCap
Com::Objsys::Csta::Devices::SiemensCap, 62
SiemensHicom300
Com::Objsys::Csta::Devices::SiemensHicom300, 63
SiemensHipath3000p2
Com::Objsys::Csta::Devices::SiemensHipath3000p2, 64
SiemensHipath3000p3
Com::Objsys::Csta::Devices::SiemensHipath3000p3, 65
SiemensHipath4000
Com::Objsys::Csta::Devices::SiemensHipath4000, 66
SiemensRealitis
Com::Objsys::Csta::Devices::SiemensRealitis, 67
SingleStepTransfer
Com::Objsys::Csta::Phase3::GenericCSTAp3, 36
SnapshotCall
Com::Objsys::Csta::Phase3::GenericCSTAp3, 37
SnapshotDevice
Com::Objsys::Csta::Phase1::GenericCSTAp1, 20
Com::Objsys::Csta::Phase2::GenericCSTAp2, 27

Com::Objsys::Csta::Phase3::GenericCSTAp3, 37
StartDataPath
Com::Objsys::Csta::Phase3::GenericCSTAp3, 37
StatusCode
Com::Objsys::Csta::Common::CSTAResponseInfo, 14
StatusMessage
Com::Objsys::Csta::Common::CSTAResponseInfo, 14
StopDataPath
Com::Objsys::Csta::Phase3::GenericCSTAp3, 37
StopRing
Com::Objsys::Csta::Phase3::GenericCSTAp3, 38

TadiranCoral
Com::Objsys::Csta::Devices::TadiranCoral, 70
ThreadContext
Com::Objsys::Csta::Phase1::GenericCSTAp1, 21
Com::Objsys::Csta::Phase2::GenericCSTAp2, 28
Com::Objsys::Csta::Phase3::GenericCSTAp3, 39
TotalLength
Com::Objsys::Csta::Common::SocketState, 69
TransferCall
Com::Objsys::Csta::Phase1::GenericCSTAp1, 21
Com::Objsys::Csta::Phase2::GenericCSTAp2, 28
Com::Objsys::Csta::Phase3::GenericCSTAp3, 38
TransferFromDevice
Com::Objsys::Csta::Phase3::SingleStepTransferInfo, 68
TransferringCallID
Com::Objsys::Csta::Phase3::SingleStepTransferInfo, 68
TransferToDevice
Com::Objsys::Csta::Phase3::SingleStepTransferInfo, 68

UnifyOpenscapeX5
Com::Objsys::Csta::Devices::UnifyOpenscapeX5, 72

WaitForROSEResponse
Com::Objsys::Csta::Common::PBXSession, 52