Document Identity
f92e8c8d-d3b0-47e6-b009-41acf714b5e9

Date	Rev.	
2002-03-12	43	

CTNG™ v1.0 Reference Manual

Prepared	Subject Responsible	Document Identity		
Mikica B Kocic	Mikica B Kocic	f92e8c8d-d3b0-47e6-b009-41acf714b5e9		
Approved / Checked		Date	Rev.	File
-		2002-03-12	43	-

CTNG™ Version 1.0
Reference Manual

NOTICE

All contents of this document are subject to change without notice and do not represent a commitment on the part of IPTC AB. Every effort is made to ensure the accuracy of this information. However, due to ongoing product improvements and revisions, IPTC AB cannot guarantee the accuracy of this material, nor can it accept responsibility for errors or omissions. No warranties of any nature are extended by the information contained in these copyrighted materials. Use or implementation of any one of the concepts, applications, or ideas described in this document or on web pages maintained by IPTC may infringe one or more patents or other intellectual property rights owned by third parties. IPTC does not condone or encourage such infringement. IPTC makes no warranty with respect to such infringement, nor does IPTC waive any of its own intellectual property rights which may cover systems implementing one or more of the ideas contained herein. Procurement of appropriate intellectual property rights and licenses is solely the responsibility of the system implementer. The software referred to in this document is provided under a Software License Agreement. Refer to the Software License Agreement for complete details governing the use of the software.

All names, products, and services mentioned herein are the trademarks or registered trademarks of their respective organizations.

CTNG™ is a trademark of IPTC AB.

DISCLAIMER

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT ANY EXPRESS OR IMPLIED WARRANTY OF ANY KIND INCLUDING WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT OF INTELLECTUAL PROPERTY, OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT SHALL IPTC AB OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, LOSS OF INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE MATERIALS, EVEN IF IPTC AB HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. BECAUSE SOME JURISDICTIONS PROHIBIT THE EXCLUSION OR LIMITATION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, THE ABOVE LIMITATION MAY NOT APPLY TO YOU. IPTC and its suppliers further do not warrant the accuracy or completeness of the information, text, graphics, links, or other items contained within these Materials. IPTC may make changes to these Materials, or to the products described therein, at any time without notice. IPTC makes no commitment to update the Materials.

IPTC Technology Communication AB
Ostmästargränd 12, 2TR
120 89 Stockholm, Sweden
Telephone +46 8 602 3191, Telefax +46 8 602 3199
Email: mailto:info@iptc.se
Web: http://www.iptc.se

Prepared	Subject Responsible	Document Identity		
Mikica B Kocic	Mikica B Kocic	f92e8c8d-d3b0-47e6-b009-41acf714b5e9		
Approved / Checked		Date	Rev.	File
-		2002-03-12	43	-

TABLE OF CONTENTS

Prepared	Subject Responsible	Document Identity		
Mikica B Kocic	Mikica B Kocic	f92e8c8d-d3b0-47e6-b009-41acf714b5e9		
Approved / Checked		Date	Rev.	File
-		2002-03-12	43	-

LIST OF FIGURES

LIST OF TABLES

Introduction

Overview

What Is the CTNG?

The CTNG is a software module that allows a computer to interact with a PABX using an interface based on CSTA. The software module is not a complete application that can be used directly, but is rather a means for writing customized applications. The CTNG consists of routines for managing CSTA objects, and performing services based on the CSTA Phase I standard.

What Is CSTA?

CSTA stands for Computer-Supported Telecommunications Applications and is a standard for interconnecting general purpose computers with telephony exchanges. The purpose of CSTA is to enhance the functionality of both the computers and the telephony exchange. This is achieved by allowing them to use the services provided by the other part, without significantly changing or redesigning any of them.

CSTA defines the interact mechanisms between computers and telephony exchanges independently of their physical implementation. This makes it possible for any computer to connect and interact with any exchange, providing that both support CSTA.

About This Document

This document is a guide for how to use the CTNG in order to write an application on a computer that uses the CSTA services provided by a switch. The first chapters deal with general information, while the last chapters are intended for programmers and deal with programming details.

What You Need To Know?

In order to use the CTNG you need to have experience with Microsoft Component Object Model (COM), and a programming language that support COM.

Other Recommended Reading

See References on page 16.

Overview of the Ericsson MD110 PABX System

The Ericsson MD110 is a full featured, stored program controlled, digital switching system that supports simultaneous voice and data transmission over a single pair of wires.

The Ericsson MD110 system is a redundant, easily expandable system. Redundant means that all vital components, such as the control system, are duplicated. Easily expandable system means that it can grow easily due to its modularity. In order to expand a system you simply add more hardware units to it. These features together with the utilization of distributed architecture provide maximum system availability.

The Ericsson MD110 modular architecture consists of two basic building blocks: the Line Interface Module (LIM) and the Group Switch (GS). A LIM can function as an autonomous system or as an integral part of a network.

A system that requires only two LIMs can be interconnected directly. For systems that require three or more LIMs, a GS is required to interconnect them.

Ericsson MD110 reliability is attained through distributed processing and modular redundancy. Since the processing power is distributed to all the LIMs, a hardware or software fault impacts service only within the LIM where the fault occurs. The rest of the system is unaffected.

The system supports automatic supervision and fault handling. Errors and faults in any of the hardware or software units are reported as alarms.

In order to operate the system, a Man-Machine interface is provided. This interface allows a system operator to execute commands for reconfiguring, supervising and controlling the system.

The basic Ericsson MD110 provides all common telephony functions expected on a modern MD110. Features such as Follow Me, Abbreviated Dialing, and Last Number Redial are implemented for efficiency and productivity.

Two separate applications provide additional functionality for use in trading and control environments respectively. These two applications can accommodate various types of operator consoles: color and monochrome plasma panels or hard-key terminals.

MD110 supports integration with various computer systems using an interface based on the ECMA-CSTA open standard. Its name is ApplicationLink and it allows the MD110 to support any CSTA compliant application or API. MD110 ApplicationLink also supports Microsoft’s proprietary TAPI interface.

Prepared	Subject Responsible	Document Identity		
Mikica B Kocic	Mikica B Kocic	f92e8c8d-d3b0-47e6-b009-41acf714b5e9		
Approved / Checked		Date	Rev.	File
-		2002-03-12	43	-

CSTA

Purpose of CSTA

Computer-Supported Telecommunications Application (CSTA) is supported by a computing component (normally based in the computing network) and a switching component (normally based in the telecommunications network). The goal of CSTA is to define and enhance the interworking capabilities and functions of the computing and switching networks independent of any physical implementations. Ideally, services provided by one network are incorporated into the other. The benefit of such a relationship is that new applications can be supported without major redesign or modification.

The CSTA Standard provides a protocol that enables computing and switching vendors/customers to enhance network capabilities using Computer Telephony Integration (CTI) technology.

CSTA Standards

The CSTA Standard defines the standard CSTA Services which is used by all switch vendors for CSTA implementation. The standard provides general descriptions of design and implementation of telephony services. The standard is intended to provide CSTA application programmers with enough information to ensure that all switch and computer vendors support the same set of telephony services independent of platform types.

The standard, however, does not provide details to support programming interfaces and telecommunication integration for specific platforms. Though CSTA Services of each vendor's switches should work the same way, typically, individual switches are implemented with different limitations.

The CSTA Standard went through several versions (named Phases). Relationship between phases is shown on the following figure.

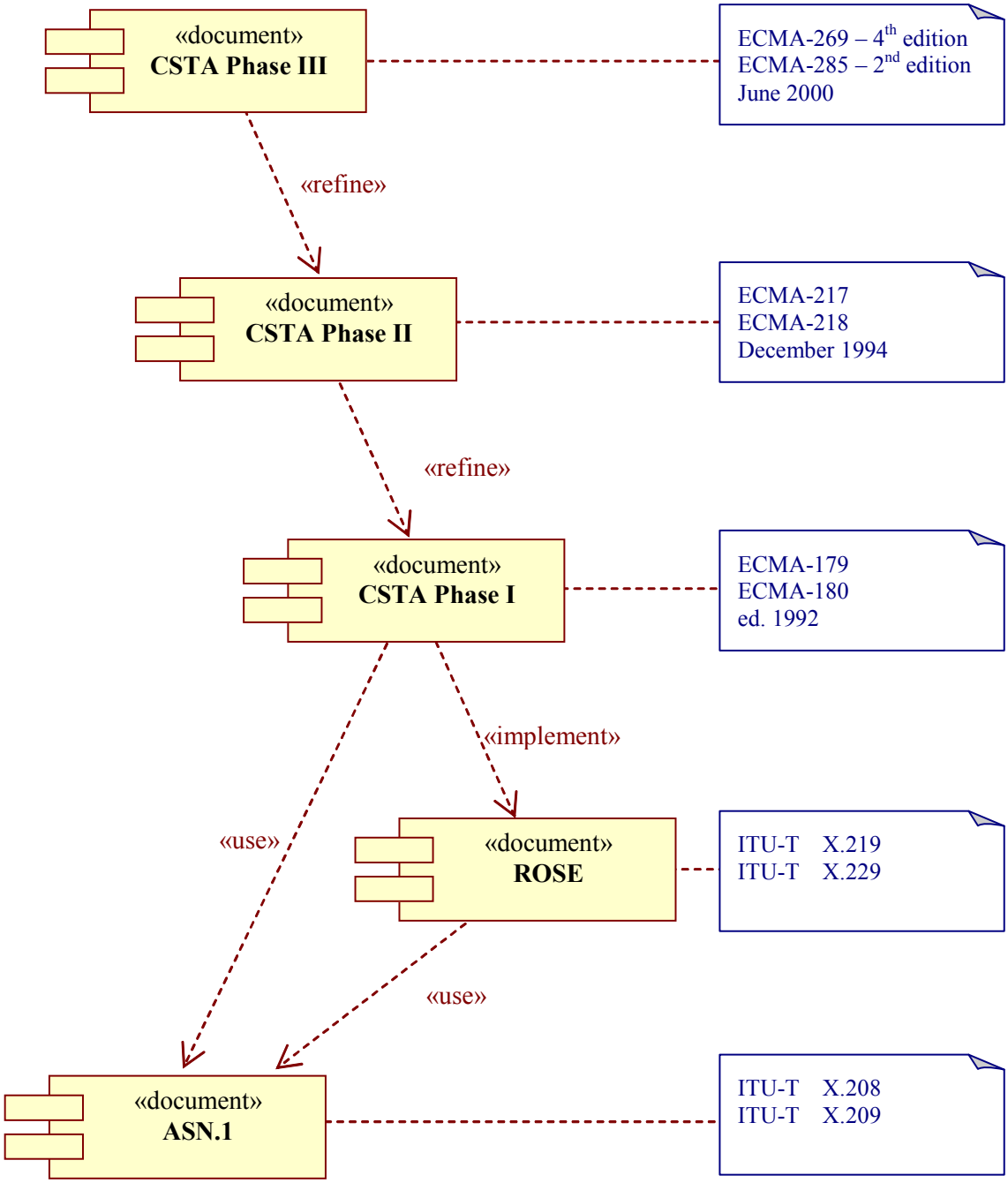

Figure 1 – CSTA Standard; Component View Diagram

CSTA-Specific Definitions

The prefix CSTA applies to all terms included in the following list:

ACD Agent

A CSTA user that is a member of an inbound or outbound ACD group. ACD Agents are distinguished from other CSTA users by their ability to sign-on (Login) to systems that coordinate and distribute calls.

Active Call

A CSTA call for which the local (or subject) Connection is in the Connected State.

Agent

A CSTA user authorized to act on behalf of the provider of the CSTA application.

Alerting Call

A CSTA call for which the local (or subject) Connection is in the Alerting State.

Application

A cooperative process between a Switching Function performed within a switching network and a Computing Function performed within a computing network.

Application Domain

The union of one switching sub-domain and one computing sub-domain.

Basic Call

A call involving exactly two associated devices.

Call

A Switching Function communications relationship generally between two or more devices. In some circumstances, including set-up and release, there may be only one device. A call is a CSTA Object.

Complex Call

A call involving more than two devices.

Computing Domain

The set of computers and their objects that can be reached by a CSTA application from a switching domain.

Computing Function

The part of the domain needed to support CSTA applications that is also within a Computing or Special Resource sub-domain.

Computing Sub-domain

Any configuration of inter-connected computers that presents the appearance and functionality of a single computer to the switching and special resource domains.

Connection

An object defined by CSTA to represent the relationship between a call and a device.

Connection Identifier

An identifier used in CSTA to identify a relationship between a specific call and a specific device. The CSTA Connection Identifier comprises a Call Identifier and a CSTA Device Identifier. Together, these identifiers specify a unique CSTA Object in the context of a CSTA Association.

Device

A logical entity and CSTA Object that serves as an endpoint for a call and accepts □ignaling information from, and provides such information to, the Switching Function. A device can encompass multiple endpoints that act together (forming a group) to provide this function.

Directory Number

A logical representation of a party or device. It is typically associated with a line (extension) circuit.

Domain

The union of the switching, computing and special resource domains.

Event

A stimulus that causes a change in the state of a CSTA object.

Event Report

A message that indicates a change in the state of a CSTA object.

Held Call

A CSTA call for which the local (or subject) Connection is in the Hold state.

Interconnection Service Boundary

The abstract Service Boundary within a system supporting a CSTA Application that separates the communications component of the application from the networking support functions of the system.

Object

An abstract entity that embodies some aspect of the externally-visible and/or functional characteristics of a physical entity.

Party

An entity outside the Switching Function that has the ability to use the Switching Function.

Security

The characteristics of a system that give it resistance to misuse and unauthorized access.

Security Domain

A bounded set of security entities that is subject to a single security policy and a single security administration.

Security Object

A passive entity to which a security policy applies.

Security Policy

A set of rules that specify the procedures and mechanisms required to maintain the security of a system.

Security Service

A set of operations designed to support some aspect of security in a system.

Security Subject

An active entity to which a security policy applies.

Service

A benefit provided by one CSTA application process to another.

Service Boundary

The functional boundary between a CSTA Computing Function and a CSTA Switching Function as it is established via their Interconnection Service Boundaries over some underlying interconnection medium.

Special Resource Domain

The set of special resources and their objects that may be reached directly or indirectly by a CSTA application from a computing or switching domain.

Special Resource Function

That part of the domain needed to support CSTA applications implemented within a special resource network or sub-domain.

Special Resource Sub-domain

Any configuration of inter-connected special resources that presents the external appearance and functionality of a single special resource to the computing or switching domain.

State

An indication of an object's current condition that permits prediction of the object's future behaviour.

Switching Domain

The set of switches and their objects that may be reached by a CSTA application from a computing or special resource sub-domain.

Switching Function

The part of the domain needed to support CSTA applications that is implemented within a switching sub-domain.

Switching Sub-domain

Any configuration of inter-connected switches that presents the functionality of a single switch to the computing or special resource domain.

User

A person, process or piece of equipment that receives direct benefit (e.g. added functionality, improved performance) from services provided by a CSTA application.

Acronyms

ACD	Automatic Call Distribution
ACSE	Association Control Service Element
API	Application Programming Interface
ASE	Application Service Element
ASN	Abstract Syntax Notation
BRI	Basic Rate Interface
COM	Component Object Model
CSTA	Computer-Supported Telecommunications Applications
ECMA	International industry association founded in 1961 and dedicated to the standardization of information and communication systems.
ID	Identifier
ITU-T	International Telecommunication Union – Telecommunication Standardization Section
ISDN	Integrated Services Digital Network
ODP	Open Distributed Processing
OSI	Open Systems Interconnection
PABX	Private Automatic Branch Exchange
PAC	Privilege Attribute Certificate
PDU	Protocol Data Unit
PRI	Primary Rate Interface
PTN	Private Telecommunications Network
ROSE	Remote Operation Service Element
SIT	Special Information Tone

References

ECMA-179 Services for Computer Supported Telecommunication Applications (CSTA) Phase I

ECMA-180 Protocol for Computer Supported Telecommunication Applications (CSTA) Phase I

ECMA-217 Services for Computer Supported Telecommunication Applications (CSTA) Phase II

ECMA-218 Protocol for Computer Supported Telecommunication Applications (CSTA) Phase II

ECMA-138 Security in Open Systems - Data Elements and Service Definitions

ECMA TR/46 Security in Open Systems - A Security Framework

ECMA TR/52 Computer Supported Telecommunications Applications (CSTA)

ECMA TR/68 Scenarios for Computer Supported Telecommunications Applications (CSTA) Phase II

ISO 7498 Information processing systems - Open Systems Interconnection - Basic reference model

ISO 8649 Information processing systems - Open Systems Interconnection - Service definition for the Association Control Service Element (this corresponds to **ITU-T Rec. X.217**)

ISO/IEC 8824 Information technology - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1) (this corresponds to **ITU-T Rec. X.208**)

ISO/IEC 8825 Information technology - Open Systems Interconnection - Specification of basic encoding rules for Abstract Syntax Notation One (ASN.1) (this corresponds to **ITU-T Rec. X.209**)

ISO/IEC 9072-1 Information processing systems - Text communication - Remote operations - Part 1: Model, notation and service definition (this corresponds to **ITU-T Rec. X.219**)

ISO/IEC 9072-2 Information processing systems - Text communication - Remote operations - Part 2: Protocol specification (this corresponds to **ITU-T Rec. X.229**)

ISO/IEC 10031-1 Information technology - Text and office systems - Distributed-office-applications model - Part 1: General model

AL40PG Ericsson MD110 Application Link Version 4.0 (AL40) Application Programmer's Guide, Doc-Id: **EN/LZT 102 2273 R5A**

CSTA Functional Architecture

The functional architecture described in ECMA TR/52[1] is summarized in this section. The objective of CSTA Architecture is to define the inter-working mechanisms among Computing, Switching and Special Resource Functions independently from their physical implementations. The concepts of distribution of Computing, Switching and Special Resource Functions, CSTA Service, client server model, and CSTA objects as abstracted at a CSTA Service Boundary, will now be introduced.

A CSTA application is supported by two or more of the following:

- a computing component (normally based in the computing network),
- a switching component (normally based in the telecommunications network),
- a special resource component (normally associated with either the computer or telecommunications network).

The operation of these components involves one or more interactions among them.

Distribution of Computing and Switching Functions

Typically, the Computing Functions are implemented by one or more computers in a computing network, the Switching Functions are implemented by one or more switches in a telecommunications network and the Special Resource Functions are implemented by one or more physical devices co-resident with either the Computing or Switching Function. It is possible, however, for some Computing Functions to be performed within the Switching Function and some Switching Functions within the Computing Function. Similarly, Special Resource Functions might be performed within either the Computing or Switching Functions.

The CSTA application appears to the user (human or machine) as a single application on a single network, not as two or more separate functions on two or more separate networks (as it is, in fact, implemented).

Since the functions of the CSTA applications are (in most situations) distributed, some form of communications support is required. This can be shown by expanding each of the distributed functions into a processing component – or application functionality (to support the defined interactions), a communications component – or CSTA Services (to support the necessary exchange of messages), and networking support – or a lower-layer interconnection service provider. The relationship is shown in the following figure.

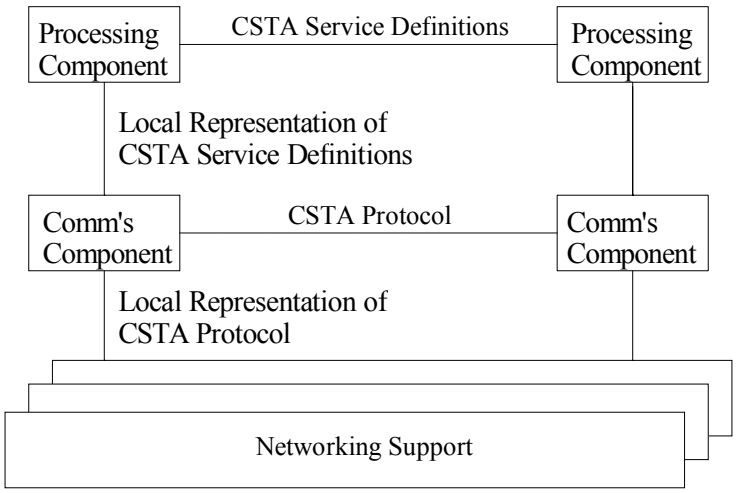

Figure 2 – Functional Diagram of the relationship between CSTA Elements

It can be seen in Figure 2 that the distributed application functions interact with their peers in accordance with CSTA Service definitions. In CSTA Standard, *Service Descriptions* define these interactions and the service interface between the application functionality and the local CSTA Service by which the peer-to-peer service interaction is supported. A CSTA Service communicates with its peer using CSTA protocol (i.e. the set of messages and associated sequencing rules etc. defined in CSTA Standard). Note, however,

[1] ECMA technical committee for Computer Supported Telecommunication Applications

that the CSTA protocol is designed to support various approaches and, as a consequence, some protocol elements are optional and their use is implementation-dependent.

In an OSI environment the application functions and CSTA Service form an application process invocation. The necessary communications component is provided by an application entity invocation considered to reside in the OSI application layer. The underlying networking support is typically provided by OSI lower layers.

CSTA Service

In the context of the OSI Reference Model and excluding the Application layer, the word “service” refers to the benefit provided by one layer to its adjacent, higher layer.

In the context of the ITU-T definition of the services provided by a real network, e.g. an ISDN, the term “service” applies to that which is offered by the network to a user at a given reference point, e.g. the S reference point.

How the OSI layer and ITU-T network notions of “service” relate to one another is shown, in simplified form, in the following figure. OSI layer services have a vertical orientation - i.e. each layer provides an OSI service to the layer above it. ITU-T ISDN Basic and Teleservices, as the latter also embrace those of the Application layer, have a horizontal orientation - i.e. service is provided between peers in the same OSI level. In the following figure, “OSI Services” are provided by the Lower Layer Interconnection System to the Application Layer, while “ITU-T/CSTA Services” are provided between the Switch Services and Computer Services Application Layer Functions.

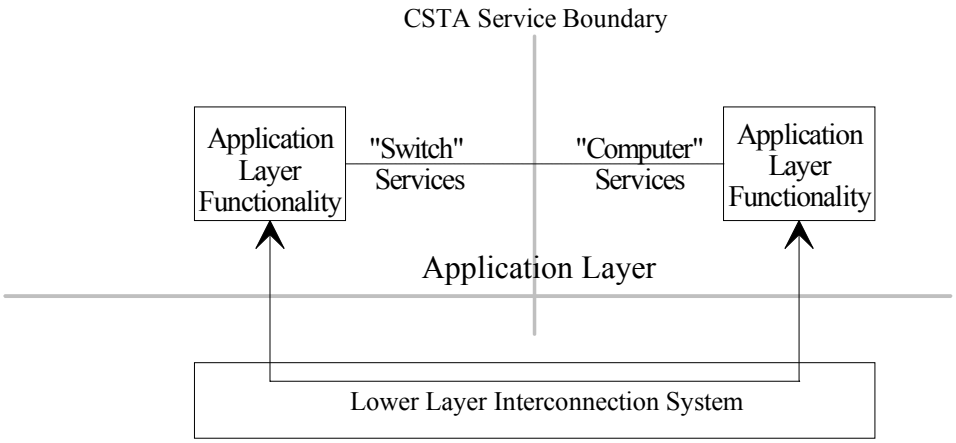

Figure 3 – OSI and ITU-T/CSTA uses of the term “Service”

Unless otherwise qualified, in CSTA Standard the term “Service” means the benefit provided by one application layer process to its peer application layer process(es).

CSTA Services have been designed to allow decoupling from the actual Telecommunications Services provided to users by the Switching Function. This makes CSTA independent of the specific user-to-network interface of the particular terminals for which CSTA is requesting services. The Switching Function is therefore responsible for determining how to support a given CSTA request. The relationship between CSTA Services and services provided to users by computers and Special Resources is similarly de-coupled: computers and Special Resources (such as voice synthesizers, recorders, etc.) may provide CSTA Services in any way that is convenient and/or appropriate. CSTA imposes no implementation constraints on the internal functioning of computer applications, Special Resources or their connections to the Switching Function.

For example, CSTA does not specify how to provide the Make Call Service on terminal types such as analog, ISDN, etc. It is expected that the Switching Function will use existing telecommunications services when providing CSTA Services on terminals for which standards exist (e.g. when CSTA requests Hold Call Service for a PTN terminal, the Switching Function will respond to that request according to the Supplementary Service Call Hold [SS-HOLD] Service definition).

CSTA has no knowledge of how the Switching Function accomplishes requested CSTA Services. The Switching Function provides only an abstraction to the CSTA Service Request or of how the Service is realized.

Per-Service Client/Server Model

The communications (as opposed to the processing) mechanism required to support the CSTA application can be modelled as a client/server relationship (such as described in ISO/IEC 10031-1). A processing component (identified in ISO/IEC 10031-1 as the User) requests a Service. Its local communications component, termed a client, invokes that particular service by communication with its peer, termed a server. The client/server relationship models application level communication and hence can be considered as belonging to the OSI application layer.

Because the CSTA architecture provides bi-directional capabilities among the Switching, Computing and Special Resource Functions, the client/server relationship is possible in both directions between any pair of functions as depicted (for the Computing and Switching Functions) in the following figure. Similar representations could also be drawn relating the Special Resource Function to the Computing and Switching Functions. Note that in this figure, the arrows show the direction of service provision - not the direction of request for service.

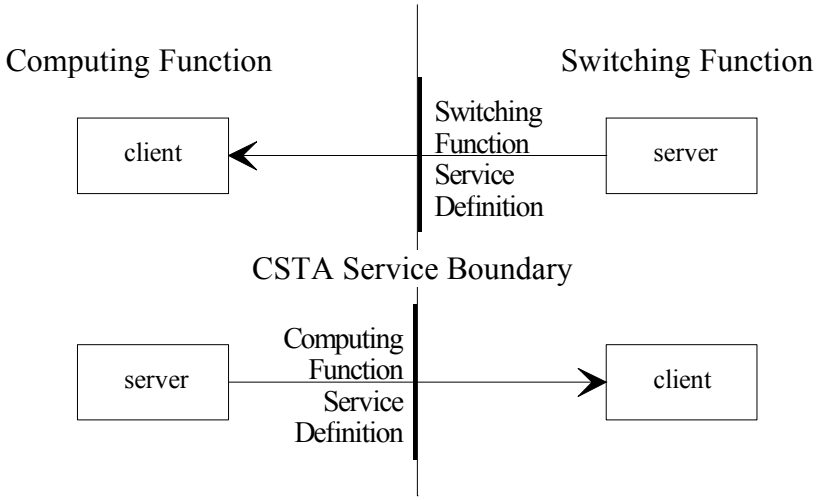

Figure 4 – Bi-directional Services Illustration

Services in which the Computing Function is the client and the Switching Function is the server are defined as Switching Function Services. An example of a Switching Function Service is the Make Call Service.

Services in which the Switching Function is the client and the Computing Function is the server are defined as Computing Function Services. An example of a Computing Function Service is the Route Request Service.

Service definitions in which the Special Resource Function is the server and the Computing or Switching Function is the client are defined as Special Resource Function Service definitions. An example of a Special Resource Function Service is the Play Message Service.

Service and Objects

Services provided by a server to a client consist of observing and/or acting upon objects that can be accessed by the server on behalf of the client. The objects and their behaviour, as perceived over the client/server interface, are defined in implementation-independent terms in the CSTA operational model (see clause 6, CSTA operational model).

CSTA Operational Model

The operational model considered for CSTA is summarized in this clause. For the purposes of standardization, all definitions and procedures specified in this clause are normative (mandatory). The modelling aspects are informative.

The set of accessible Computing, Switching and Special Resource Functions from which an application might receive service defines a CSTA domain. An example of a CSTA domain is shown in the next figure. The CSTA domain contains switching, computing and special resource domains that are divided in the figure by the heavy lines. The special resource, switching and computing domains comprise Computing Functions (C1, C2 and C3), Switching Functions (S1, S2 and S3), and Special Resource Functions (SR1, SR2 and SR3). Each function can provide to a CSTA application, a view of the domain in which the function resides. Each such view defines a sub-domain. If one or more functions provide an identical view, then these functions are part of the same sub-domain. CSTA applications encompass at least two different sub-domains, and are represented in the next figure as application domains.

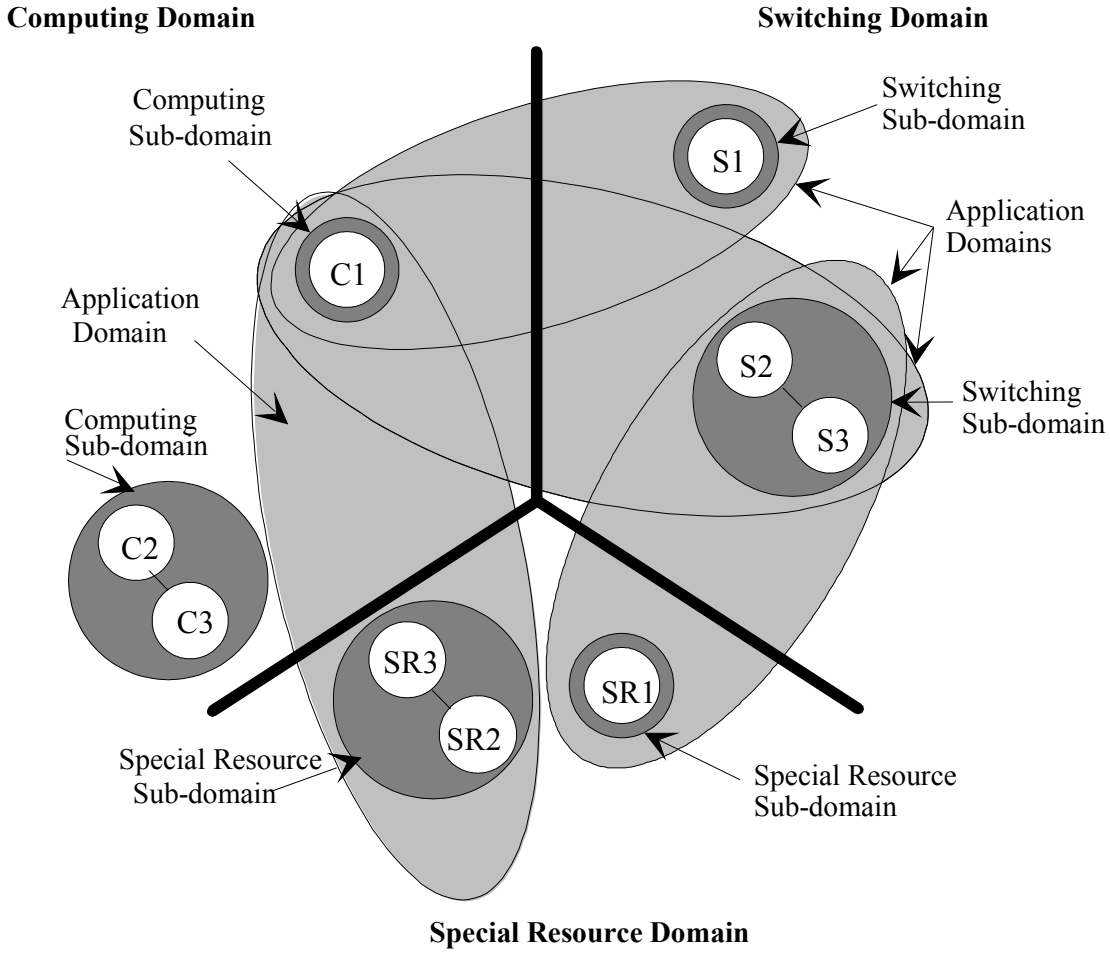

Figure 5 – Domains and Sub-Domain

Note that a function may provide a view to an application that includes not only the objects within its sub-domain, but also the objects it can view in another (presumably related) sub-domain. For example (in figure 0), a computing sub-domain {C1} may receive a view of a switching sub-domain from a switching sub-domain {S2+S3}. That switching sub-domain may receive a view of a special resource sub-domain from a special resource sub-domain {SR1}, and relay that view, in addition to the view of its sub-domain, to the Computing Function. This relay may preserve two views of separate switching and special resource sub-domains, or it may provide a combined view of a switching/special-resource sub-domain. As shown in the figure, {C1} also may have its own, direct view of a special resource sub-domain {SR2+SR3}. Finally, {C2+C3} represent a computing sub-domain that is potentially, but not yet, involved in CSTA transactions with other sub-domains because an association has not yet been established between any other sub-domain and {C2+C3}.

Switching Sub-domain Model

The tools needed to provide an abstract view of the Switching Function are defined by the switching sub-domain model. This model allows an application to conceptualize the Switching Function's operation. To provide this abstract view, CSTA defines several CSTA switching sub-domain model Objects that can be observed and acted upon by the Switching Function on behalf of the Computing Function. Those objects include Device, Call, and Connection.

Device

CSTA enables manipulation and observation of devices that allow users to access telecommunications services.

> ***Note 1:***
>
> *It is not claimed that CSTA Standard alone supports ISDN (or any other) devices because, for example, of the additional information required to support such devices in PTNs. CSTA only provides some information allowing selection among ISDN devices sharing the same directory number (bearer capability, subaddress, etc.). Another example, that applies generally to telecommunications networks (including ISDN and OSI), is specifying the originator for a call that is established via CSTA. With the current signalling support, each party in a call can act only as a called party because the "network" is acting to originate the call. This situation has implications for both the network-to-terminal signalling and any application-level signalling that is significant to the calling party (e.g. issuing A_Associate).*

Devices that are visible or controllable via CSTA are known as CSTA Devices.

CSTA devices can be either physical devices (such as buttons, lines, trunks, and stations) or logical devices (such as groups of devices, pilot numbers, and automatic call distribution groups). Devices have attributes that allow CSTA to monitor and manipulate them. The CSTA attributes of any CSTA device are:

1. **Device Type** - differing types of device can be used for various purposes and can be manipulated and observed differently within CSTA. A CSTA Device is one of the following types:

   ACD - Automatic Call Distributor (ACD) is a mechanism that queues and distributes calls within a Switching Function. An ACD device (as opposed to ACD-group device) comprises only the distribution mechanism and not devices to which the mechanism can distribute calls.

   ACD group - Automatic Call Distributor (ACD) group is the mechanism that queues and distributes calls within a Switching Function as well as the devices to which that mechanism distributes calls. As an ACD group (as opposed to ACD), the device consists both of the distribution mechanism and the devices to which the mechanism can distribute calls.

   Button - is one instance of a call manipulation point at an individual station. Simple analogue stations often have no physical buttons but behave as if they had (at least) one. Some advanced stations can emulate several analogue stations and often represent those stations by buttons. In some situations it is desirable to identify a specific button on a multi-button station. Note that a station with several buttons could have (but need not have) the same telephone number assigned to multiple buttons.

   Button group - is two or more instances of a call manipulation point at an individual station.

   Conference Bridge - is a device that automatically provides a conferencing function among the calls present at the device.

   Line - is a communications interface to one or more stations. In some situations it may be impossible to identify individual stations that share a line and a single directory number.

   Line group - is a set of communications interfaces to one or more stations.

   Operator - is a device that is used to interact with a party to assist in call setup or to provide certain other telecommunications services. This device differs from other devices in that it is often involved in setting up other calls and is not usually part of the call after the call is connected.

   Operator group - two or more operators used interchangeably or addressed identically.

   Park - is a device that is used solely for parking calls. Note that calls also may be parked at other types of devices.

   Station - is the traditional telephone device, either simple or "featured". A featured station is a physical unit with one or more buttons and one or more lines.

Station group - is two or more stations used interchangeably or addressed identically.

Trunk - a device used to access other switching sub-domains. In order to manipulate and view calls that cross a CSTA switching sub-domain it may be desirable to address the point at which the call crosses the boundary. This point is generally a trunk or trunk group. A trunk is an individual circuit.

Trunk group - a set of independent trunks. Often, many trunks provide connectivity to the same place. These trunks are frequently grouped together and are all accessed using the same identifier. In such a configuration the individual trunks are used interchangeably. Note that the set may consist of only a single member trunk.

2. **Device Profile** - differing types of device can be observed and manipulated differently within CSTA. These types have distinct Device Profiles corresponding to their capabilities and characteristics. CSTA represents these types by Device Class augmented with ISDN Setup information where appropriate.

   Device Class - a CSTA Device belongs to at least one, and may belong to more than one, of the following classes:

   Data - devices in this class are used to make digital data calls (both circuit-switched and packet-switched). This class includes digital computer interfaces and G4 facsimile machines.

   Image - devices in this class are used to make digital data calls involving imaging, or high-speed, circuit-switched data in general. This class includes digital video telephones and CODECs.

   Other - a class comprising devices not in the Data, Image, Audio or Voice classes.

   Voice - devices in this class are used to make speech calls. This class includes standard telephones.

   Audio - 3.1 KHz audio. Devices in this class are used to make audio calls excluding speech calls. It includes G3 fax and facsimile machines.

   Additional ISDN type information such as the following also may be included to help distinguish devices:

   - Bearer Capability,
   - Subaddress (for both calling and called devices),
   - High Layer compatibility,
   - Low Layer compatibility.

   These information elements are defined in ISO/IEC 11572: 1993.

3. **Device Identifier** - each device that can be observed and/or manipulated must be referenced across the CSTA Service Boundary. To accomplish this, each CSTA device is identified using a CSTA Device Identifier. Device Identifiers may be static or dynamically-assigned.

   A static Device Identifier is stable over time. It remains constant and unique between calls, associations and within both the switching and computing sub-domains. An example of a static CSTA Device Identifier is an E.164 Directory Number.

   It may be useful for the Switching Function to convert a Device Identifier to another static form for use in service interactions. An example, it might be useful to transform a Public Directory Number into a Private Directory Number. This transformation allows service interactions to be independent of the identification mechanism and allows reduction in the amount of data exchanged. This shortened form of Device Identifier is known as a CSTA Short Form Device Identifier.

   A static CSTA Device Identifier may be used in conjunction with the CSTA Device Profile in order to distinguish among devices that share a Device Identifier.

   A dynamically-assigned Device Identifier is temporary (lasting for the duration of a call) and may be created at any appropriate time. Once a CSTA device has been included in a call, it may be desirable to continue to refer to the particular instance of the device associated with this call for manipulation or tracking. A static Device Identifier may not always be sufficient because it may not be available or because it is too long and cumbersome for efficient use. In these cases the Switching Function can dynamically assign a Device Identifier as a device reference or handle for the duration of the call. Management of the dynamically-assigned Device Identifier is discussed in 6.1.7, Management of dynamically-assigned Identifiers.

4. **Device State** - the set of Connection states that are associated directly with a particular device. Connection states are discussed further in 6.1.3, Connection.

Call

Call behaviour, including establishment and release, can be observed and manipulated across the CSTA boundary. During some call phases (e.g. establishment and release) the call is not completely formed and there may be only a single device involved (for example, the device on whose behalf the call was initiated). In many operations, such as conference and transfer, one device in a call is replaced with another device or two calls are merged into a single call. The CSTA call attributes are:

1. **Call Identifier** - a Call Identifier is a handle or reference associated with a CSTA call whereby that call can be known to, and identified by, CSTA Switching, Computing and Special Resource Functions through the call's life. A Call Identifier is allocated to each call by the Switching Function when the call first becomes visible across the CSTA Service Boundary. It may or may not be globally unique among all calls within a switching sub-domain, but it is always globally unique within the call and includes all end-points of the call. A Call Identifier can be assigned to a call before that call is fully established. For example, an incoming call may be assigned a Call Identifier when the called device is Alerting and before the call has been answered. A Call Identifier is not only reference the entire call within the sub-domain but references also the entire call outside the sub-domain (including all end points that are visible to CSTA.)

   A CSTA call can pass through various stages involving many and different devices before it finally terminates. Examples of CSTA Services that cause this are Transfer and Conference. During the operation of CSTA Services, or as a result of manual intervention, the Call Identifier may change, but the call continues as a CSTA object. Management of Call Identifiers is described in 6.1.7, Management of dynamically-assigned Identifiers.

2. **Call state** - the set of Connection states for those Connections comprising a call. Call state is returned by the Snapshot Device Service for devices that have calls. Connection states are further described in 6.1.3, Connection. Call states are described in more detail in 6.1.5, Call states.

3. **CSTA Application Correlator** - data provided by the Computing Function and associated with the call for its entire duration or until overwritten with new data. This data survives Conference and Transfer and can be provided on various events. An application may remove the Application Correlator Data by overwriting existing data with null data. If an existing call that has Application Correlator Data is Conferenced or Transferred with a consulting call that also contains Application Correlator Data, then the data from the consultation call is retained and overwrite the other data.

Connection

A Connection is a relationship between a call and a CSTA device. Note that this definition of "Connection" differs from those used in other standards. This Connection relationship can be both observed and manipulated. In fact, observation and manipulation of these Connections make up many CSTA Services (e.g. Hold Call Service, Reconnect Call Service, and Clear Call Service). Connections are CSTA Objects that have the following attributes:

1. **Connection Identifier** – a tuple of the CSTA Call Identifier and CSTA Device Identifier. For a call, there are as many Connection Identifiers as there are associated devices, and for a device there are as many Connection Identifiers as there are associated calls. The CSTA Connection Identifier is unique within a sub-domain and within a single association. As provided by the Switching Function, the Connection Identifier includes both a Device Identifier and a Call Identifier. The Device Identifier within a Connection Identifier may be dynamically-assigned. As provided by the Computing Function to the Switching Function, a Connection Identifier includes at least a Device Identifier. It may be rejected by the Switching Function if a Call Identifier is missing. If a Connection Identifier sent by the Computing Function includes only a Device Identifier, then that Device Identifier is not be dynamically-assigned. These conditions ensure that it is possible to use only a Device Identifier (without a Call Identifier) to provide a Connection Identifier. Different associations may have distinct CSTA Connection Identifiers for the same Connection. The definitions of a Connection Identifier and those identifiers that it comprises (CSTA Call and Device Identifiers) restrict CSTA Computing Functions from fabricating Connection Identifier. The Computing Function cannot use a Call Identifier until a Connection Identifier containing that Call Identifier has been received from the Switching Function.

2. **Connection state** - one of the CSTA states a Connection may have. Connection states may be reported by Snapshots on either calls or devices, and changes in Connection states may be reported

as Event Reports by Monitors. A Connection state involves a single Call/Device relationship. A simplified Connection state model is shown in the following figure.

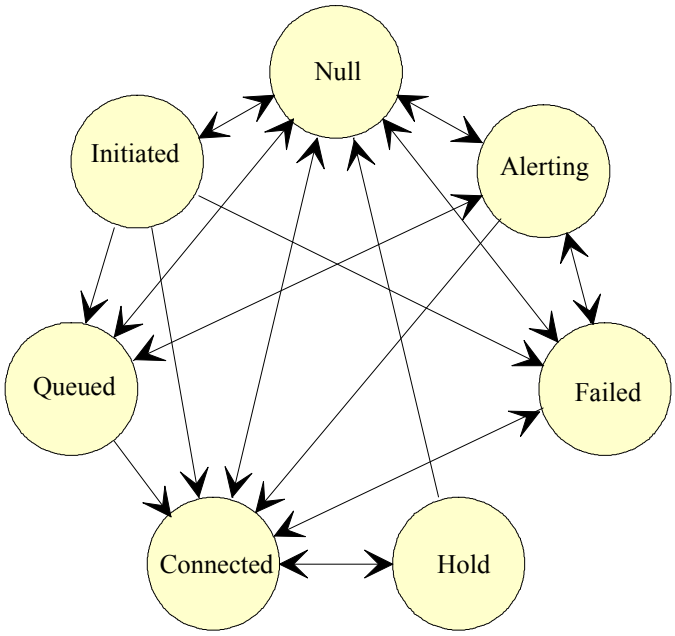

Figure 6 – Connection State Model

The states (represented by circles) presented in Figure 6 comprise the set of CSTA Connection states. State transition possibilities, represented by arrows, form the basis for providing Event Reports when such transitions occur. CSTA Connection states are not equivalent to ISDN access states. CSTA Connection states are derived from the state machine on only one side of an ISDN access. CSTA Connection states are defined as follows:

Null - A state in which there is no relationship between a call and device.

Initiated - A state in which a device is requesting service. Often this is referred to as the “dialling” state.

Alerting - A state in which a device is alerting (e.g. ringing). This indicates an attempt to connect a call to a device.

Connected - A state in which a device is actively participating in a call. This state includes logical participation in a call as well as physical participation (i.e. a Connected device cannot be on Hold).

Hold - A state in which a device is inactively participating in a call. This state includes logical participation in a call while physical participation is suspended.

Queued - A state in which call progression has been stalled. This state generally refers to two conditions but can refer to others. One condition is when a device is trying to establish a Connection with a call and this process stalls. The second condition is when a call tries to establish a Connection with a device and that process stalls.

Failed - A state in which call progression has been aborted. This state generally results when a device tries to become Connected to a call or a call tries to become Connected to a device and the attempt fails. The Failed state can result from failure to connect the calling device and call, failure to connect the called device and call, failure to create the call, failure when the call ends, and other reasons.

Call Event Reports

The reason for the Connection state model is to provide an abstract view of actual states and events that are communicated via underlying signalling systems. This abstract view is introduced to provide a language for describing CSTA Event Reports, states and Functional descriptions. Because of the topology of the Switching Function, the signals that report events and state changes have definite sources. Providing a telecommunications object (the Connection) that can be associated with the source of these signals helps when explaining the meaning of events and the operation of CSTA (and other) telecommunications services.

On a typical ISDN access to a network there exists a distributed state machine. One part of this distributed state machine resides in the ISDN device. Another part resides on the other side of the ISDN access. There is another similar distributed-state access machine that resides across the ISDN network at a similar device.

Using this concept, a call can be modelled as a collection of Connection state machines communicating with one another using signalling. When this communication occurs, a CSTA Event Report can be generated. In the following figure, this concept of communication between two state machines is illustrated for the case of establishing a simple call. Additionally, on each side of the figure the ISDN call states are indicated.

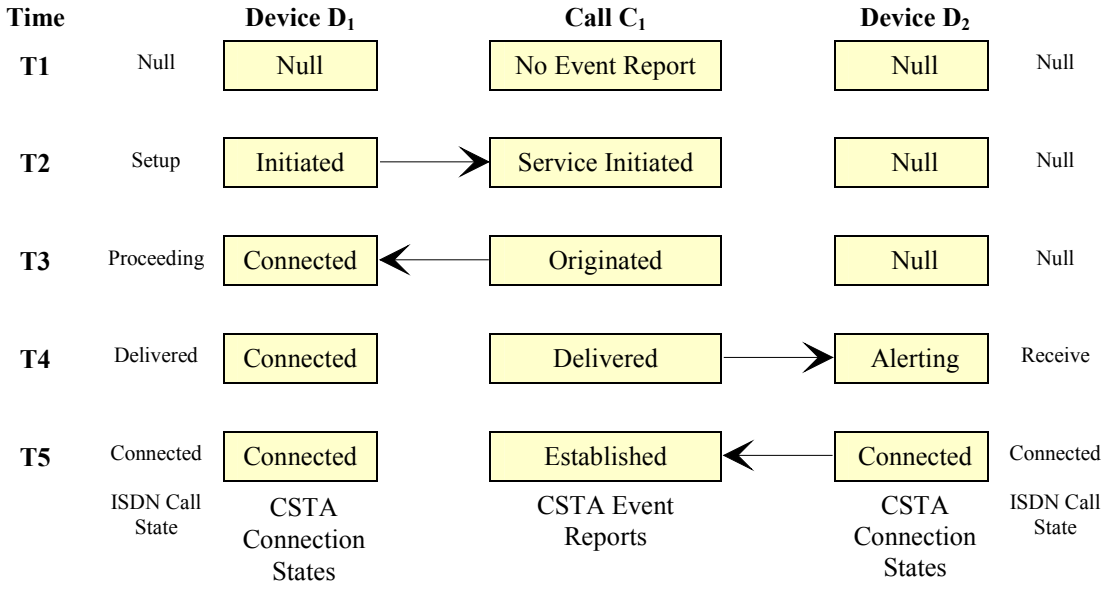

Figure 7 – Relationship of CSTA Call Event Reports

Notice in Figure 7 that the CSTA Event Reports are based on signaling interactions of the Switching Function. Many Connection events are of interest to CSTA applications. Typically, however, a CSTA application is interested in atomic telecommunications activities and these often involve many simultaneous Connection events. Generally, telecommunications operations embody changes to many Connections. These events can be summarized in a single Event Report. For instance, the Transfer, Conference and Clear Call Services all make changes to multiple Connections but are each represented by single Event Reports. The Connection state changes associated with each CSTA Event Report are defined in CSTA Standard.

Call states

The state of a CSTA Call can be precisely expressed as the list of Connection states of all the devices involved in the call. This list is called the Compound Call State. The technique of listing the Connection states to describe the Call state can describe any call state that is possible in CSTA. However, most calls involve a small number of widely-recognized states. CSTA defines those states in terms of their set of Connection states, but communicates them as atomic Call states - not as a list. These widely-recognized states are called the Simple Call States.

For calls with one known Connection state, the single Connection state is provided as a Call state.

> ***Note 2:***
>
> *Since Null can be a known Connection state, for a nascent call it is possible to have a CSTA Call state with only one non-Null Connection (see Table 1).*

For calls with more than two non-Null Connection states, the list of Connection states is provided as the call's state.

CSTA simplifies Call states by relating them (at times) to particular devices. These relationships are described by differentiating the call's Connection states. The Connection state associated with a particular device is called the local Connection state (for that device). Other Connection states are not differentiated from one another. Thus, CSTA Call state is defined for a device by the combination of Connection states as well as the order in which the Connection states are combined. For example, the Alerting-Connected

Call state is not the same as Connected-Alerting. The first is defined as Received and the second is defined as Delivered. For calls with exactly two Connections, the CSTA Call state assigned to the combinations of Connection states are summarized in the following table. If there is no Simple Call state for a particular combination of Connection states, then a Compound state is provided as the Call state. For Compound Call states, the first Connection state in the list is the local Connection state.

Table 1 - Definition of CSTA Simple Call states

Local Connection State	**Other Connection State**	**CSTA Simple Call State**
Alerting	Connected	Received
Alerting	Hold	Received-On Hold
Connected	Alerting	Delivered
Connected	Connected	Established
Connected	Failed	Failed
Connected	Hold	Established-On Hold
Connected	Null	Originated / Terminated
Connected	Queued	Queued
Failed	Null	Blocked
Hold	Alerting	Delivered-Held
Hold	Connected	Established-Held
Hold	Failed	Failed-Held
Hold	Queued	Queued-Held
Initiated	Null	Pending
Null	Null	Null

Note 3:

The Originated / Terminated state may occur both during call set-up and when the call ends. When a far-end party drops from a two-party call and the near-end end-point is not returned immediately to idle, then the Originated / Terminated state is entered for call tear-down. It is also possible to enter a blocked state when a call ends.

Agent

An Agent is a CSTA object that relates the activity of an ACD agent to an ACD device. CSTA Agents and their activities can be identified, observed and controlled using CSTA Services and Event Reports. CSTA Agent attributes are:

1. **Agent Identifier** - an Agent Identifier is allocated to each Agent by the Switching Function when each Agent first becomes visible across the CSTA Service Boundary. It may or may not be identical to the Device Identifier of the device used by the Agent.
2. **Agent Password** - an Agent password may be allocated to each Agent. The password authenticates the Agent to the CSTA Application and/or one or more of its component functions. Additional information on authentication is provided in clause 8, Security.
3. **Agent state** - a state that an Agent may take in relation to an ACD. An Agent state relates that Agent to its task of serving the queuing mechanism of a particular ACD. It is possible that an Agent has several states with respect to different ACD devices. Alternatively, an Agent may use a single state to describe its relationship to all ACD devices. Agent states are reported in Agent Event Reports. A typical transition model for Agent states is presented in the following figure.

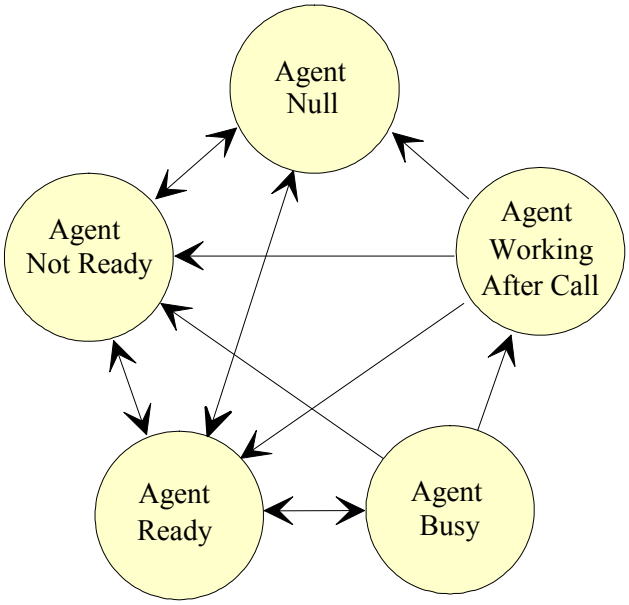

Figure 8 – Agent State Model

In Figure 8, the Agent states (circles) presented comprise the CSTA Agent state set. The transitions between states, represented by arrows, show typical states that may be entered from a given state. These transitions, when they occur, form the basis for providing Agent Event Reports.

Agent Null - the state where an Agent is not logged-on to the ACD. Logging-on and Logging-off from an ACD causes the transitions to and from this state.

Agent Not Ready - the state where an ACD Agent is logged-on to an ACD, but is not prepared to handle calls that the ACD distributes. While in this state an Agent may receive calls that are not handled by the ACD.

Agent Ready - the state where an ACD Agent is logged-on to an ACD and is prepared and waiting to handle calls that the ACD distributes.

Agent Busy - the state where a device, on behalf of an Agent, is involved with an existing ACD call at that device, even if that call is on hold at that device. Calls between agents, calls between supervisors and agents and private calls may not cause this transition.

Agent Working After Call - the state where a device, on behalf of an Agent, is no longer involved with an ACD call. While in this state an Agent cannot receive further calls from the ACD but may be performing administrative duties (e.g. updating a business order form) for a previous call.

Management of Dynamically-Assigned Identifiers

Management of dynamically-assigned Device Identifiers and Call Identifiers is provided through management of Connection Identifiers. This ensures that an identifier whose meaning is dependent on another identifier is always provided in the proper context (i.e. with the other identifier needed to resolve its meaning.) For example if a Call Identifier is given relative to a device, then giving the Connection identifier ensures that the Call Identifier is provided with its reference - the Device Identifier. Management of CSTA Connection Identifiers are provided as follows.

Connection Identifiers are provided when either a new Call is created or a new device becomes involved in a call. When a call is made a Connection Identifier is provided. A Connection Identifier is provided in Event Reports that pertain to a call. When a device becomes involved in a call, the Connection Identifier is provided in the Event Reports that occur at that device.

If a call changes its Call Identifier when a Conference or Transfer occurs, Connection Identifiers are provided to link the old Call Identifier to the new Call Identifier. Similarly, if a Device Identifier is changed, new Connection Identifiers are provided for the devices in the call.

Management of identifiers are provided via parameters included in Service acknowledgements and Event Reports.

Identifiers cease to be valid when their context vanishes. If a call ends, its Call Identifier is no longer valid to refer to that call. Similarly, if a device is removed from service or from a call, its dynamically-assigned Device Identifier becomes invalid.

Identifiers can be reused. Once an identifier has lost its context it may be re-used to identify another object. It is recommended that implementations not re-use identifiers immediately.

Individual Call and Device Identifiers are not guaranteed to be globally unique. CSTA requires that the combination of Call and Device Identifier be globally unique within a CSTA switching sub-domain. To accomplish this, either the Call Identifier, or the Device Identifier (or both) is globally unique. In many cases the Connection Identifier requires both the Call and Device Identifiers to uniquely refer to Connections in a call.

Special Resource Functions

Special Resource Functions (SRFs) are functions that are typically “added-on” to a Switching or Computing Function. They can be modelled as part of either one of the two other Functions or as something totally independent. Special Resource Functions may include various subclasses that are defined independently but share a similar relationship to the Switching and Computing Functions in the Functional Architecture and Operational Modelling. Typical SRF subclasses include Voice Units, some Conference Bridge Units, Facsimile Units and Video Units. The Voice Unit subclass is defined in CSTA Standard.

Voice Unit

A Voice Unit is a CSTA SRF and object that supports the transformation of voice between real-time and static representations (i.e. between calls and messages). Voice Units and their activities can be identified, observed and controlled using CSTA services and a state model. An example of a Voice Unit device is a Voice Mail system. CSTA Voice Unit attributes are:

1. **Call** - A Connection Identifier is used to indicate the call that the Voice Unit relates to a message. Typically the Voice Unit will record some portion of the call or play a message as some portion of a call. There are some Voice Unit Services (e.g. Delete message and Concatenate message) that deal with the control of messages and do not require an interaction with a call.
2. **Message** - a Message Identifier is used to allow manipulation of messages, many of which survive the life of multiple calls.
3. **Voice Unit state** - A state that a Voice Unit may take in relating a call with a message. A Voice Unit state relates a call to its message in terms of playing, recording, pausing, suspending, changing playback speed, etc. Voice Units may have several states concurrently with respect to different calls and messages. Voice Unit states are reported by Voice Unit Event Reports. A typical transition model for Voice Unit states is shown in the following figure.

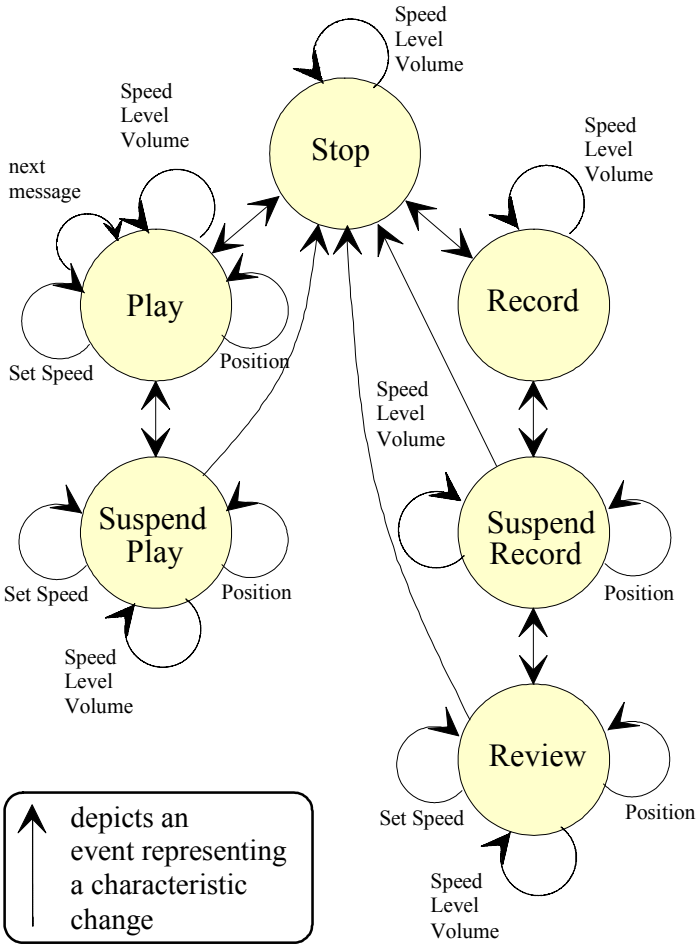

Figure 9 – Voice Unit Operational Model

In Figure 9, the states (circles) presented comprise the CSTA Voice Unit state set. Arrows represent transitions between states and show the typical states that may be entered from a given state. These transitions form the basis for providing Voice Unit Event Reports when they occur. The circular transitions show the effects of the Reposition and Set Speed Services. The following states are defined:

Stop - the state where a call and a message are not currently interacting.

Play - the state where a message delivers its information to a call.

Suspend Play - the state where a message that was in the Play state is temporarily suspended in its delivery. This state (rather than Stop) is entered when it is expected that the message will re-enter the Play state.

Record - the state where a message is created from the information in a call.

Suspend Record - the state where a message that was in the Record state is temporarily suspended from recording. This state (rather than Stop) is entered when it is expected that the message will re-enter the Record state.

Review - the state where a message that was in the Suspend Record state delivers recorded information back to the call. This allows the information provider to examine information recorded so far.

Prepared	Subject Responsible	Document Identity		
Mikica B Kocic	Mikica B Kocic	f92e8c8d-d3b0-47e6-b009-41acf714b5e9		
Approved / Checked		Date	Rev.	File
-		2002-03-12	43	-

CNTG COM Library

This document provides information regarding the CTNG CSTA Component Object Active Template Library version 1.0.

The purpose of this document is to provide an application programmer with sufficient information and detail on specific services and events that are supported by CTNGLib. This information helps the programmer in developing CTI (Computer Telephony Integration) applications that use CSTA protocol.

CTNG version 1.0 release fully supports CSTA Phase I recommendation, and cooperates with Ericsson MD110 Application Link versions 3.02 and 4.0.

Library UUID

```
library CTNGLib
uuid 3FFC9007-8F41-48F9-BE6F-D83A79BAA74A
version 1.1
helpstring "CTNG 1.1 Type Library"
```

Collections

There is one, and only one, System accessible through System object, which could contain zero or more PABX objects. Each PABX object can contain zero or more open Devices. This is shown in collection diagram in Figure 10.

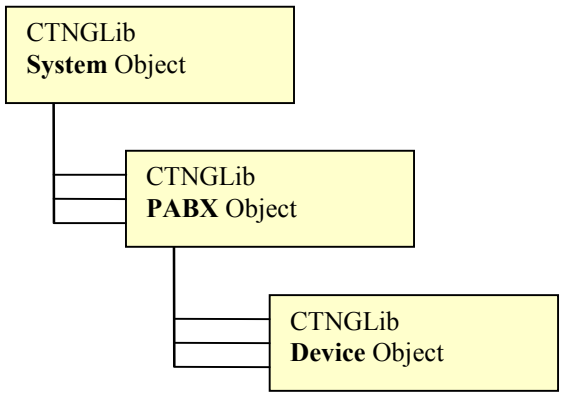

Figure 10 – CNTG COM Library Object Collections Diagram

Classes

System	The CTNGLib runtime system.
PABX	PABX / Switching domain object class.
Device	PABX's device object class.

Data Types

DeviceID	String in special text format representing device identifier.
CallID	String in special text format representing call identifier.
ConnectionID	String in special text format representing connection identifier.

Enumerated Constants

ErrorStatus	Returned by the device class CSTA services methods mostly.
SystemStatusType	CSTA system status codes.
LocalConnectionInfoType	CSTA event report local connection info codes.
EventCauseType	CSTA event cause codes.

System Class

Hierarchy

CTNGLib → **System**

Description

Provides static methods for the maintaining all CTNG objects. All objects are created as a part of one global system. In this version System objects provides two basic functions:

- Console access to service for debugging purposes using OpenConsole / CloseConsole methods, and
- AddPABX method that should be used to configure available PABXs to the system.

Interface

```
interface ISystem : IDispatch
uuid 74041709-2558-4C76-9614-63FF8E5E5452
helpstring "ISystem Interface"
```

Dispinterface

```
dispinterface _ISystemEvents
uuid F5FC4EFE-74B6-48D5-B48C-AE162F6B1A0B
helpstring "_ISystemEvents Interface"
```

Coclass

```
coclass CTNG.System
uuid B63AC033-0B40-4FD1-9BE1-5276FED1A5A1
helpstring "CTNG System Class"
```

Properties

None.

Collections

PABXs	

Methods

OpenConsole	
CloseConsole	
IsConsoleOpen	
AddPABX	

Events

None.

Remarks

None.

See Also

None.

Prepared	Subject Responsible	Document Identity		
Mikica B Kocic	Mikica B Kocic	f92e8c8d-d3b0-47e6-b009-41acf714b5e9		
Approved / Checked		Date	Rev.	File
-		2002-03-12	43	-

System.OpenConsole Method

Hierarchy

CTNGLib → System → **OpenConsole**

Syntax

```
system.OpenConsole
```

Return Value

None.

Parameters

None.

Description

This method directs the CTNG service to open and attach console to the process of system service. All internal messages are then directed through standard output and standard error handles to process console. The primary purpose of this method is to provide additional information useful for debugging purposes.

Remarks

None.

See Also

System.AddPABX Method

Applies To: System Class

System.CloseConsole Method

Hierarchy

CTNGLib → System → **CloseConsole**

Syntax

```
system.CloseConsole
```

Return Value

None.

Parameters

None.

Description

This method directs process to detach existing console from service process and to close it.

Remarks

None.

See Also

System.OpenConsole Method

Applies To: System Class

System.AddPABX Method

Hierarchy

CTNGLib → System → **AddPABX**

Syntax

```
system.AddPABX pabx_id, ip_address, tcp_port
```

Return Value

None.

Parameters

pabx_id

String, by value, input

ip_address

String, by value, input

tcp_port

Long, by value, input

Description

This method configures new PABX known to a system. Parameters to this methods are **pabx_id** as internal identifier of PABX, and CSTA (i.e. Application Link) connection point parameters given in **ip_address** and **tcp_port** form.

This method should be called prior to any CTNGLib.PABX.Open methods, because PABX.Open will used internal database information about known PABX identifiers configured by AddPABX method.

Remarks

None.

See Also

Applies To: System Class

PABX Class

Hierarchy

CTNGLib → **PABX**

Description

The PABX class is used to access configured PABX inside the System. The PABX object is a container for zero or more Device objects that can be dynamically created or destroyed. However, PABX objects are configured statically by the System.AddPABX method.

Interface

```
interface IPABX : IDispatch
uuid AE0C8561-63DD-4C52-9023-A0CCBC9EE3B3
helpstring "IPABX Interface"
```

Dispinterface

```
uuid 35588776-0A0A-4731-9C59-FD30740D6C60
helpstring "_IPABXEvents Interface"
dispinterface _IPABXEvents
```

Coclass

```
coclass CTNG.PABX
uuid 155E8B8C-6C03-470E-82F7-16F99C417A5F
helpstring "CTNG PABX Class"
```

Properties

None.

Collections

Devices	

Methods

Open	Open existing PABX in the system.
Close	Close PABX.

Events

None.

Remarks

None.

See Also

None.

PABX.Open Method

Hierarchy

CTNGLib → PABX → **Open**

Syntax

```
pabx.Open pabx_id
```

Return Value

None.

Parameters

pabx_id

String, by value, input

Description

This method opens PABX for further access given by its identifier **pabx_id**. Given identifier must be known to system prior to use (see CTNGLib.System.AddPABX method).

Remarks

None.

Example

```
Option Explicit
Public csta As New CTNGLib.System
Public pabx As New CTNGLib.pabx
Public WithEvents dev As CTNGLib.device

Private Sub Initialize
    csta.AddPABX "md110", "csta.md110.com", 2555
    pabx.Open "md110"
    Set dev = New CTNGLib.Device
    dev.Open pabx, "805"
End Sub
```

See Also

System.OpenConsole Method

Applies To: System Class

PABX.Close Method

Hierarchy

CTNGLib → PABX → **Close**

Syntax

```
pabx.Close
```

Return Value

None.

Parameters

None.

Description

This method closes already open PABX object. Normally it should not be called directly (the system will automatically purge all unused objects). This method is implicitly called when the object pointer is set to nothing.

Remarks

None.

See Also

System.OpenConsole Method

Applies To: System Class

Device Class

Hierarchy

CTNGLib → **Device**

Description

The Device class provides access to PABX device objects. PABX objects contains zero or more Device objects.

Interface

```
interface IDevice : IDispatch
uuid C391A2ED-B56A-469E-B0E3-F51D8BE88713
helpstring "IDevice Interface"
```

Dispinterface

```
dispinterface _IDeviceEvents
uuid DF33386D-07F2-40F2-8775-E136A4DFEFB0
helpstring "_IDeviceEvents Interface"
```

Coclass

```
coclass CTNG.Device
uuid 2BE420EA-C378-40F8-BF73-B19CFF854186
helpstring "CTNG Device Class"
```

Properties

None.

Methods

Open	Open device and start monitoring events on it.
Close	Close device.
AnswerCall	This operation connects an alerting call.
ClearConnection	This operation releases a specified device from a designated call.
MakeCall	This operation originates a call between two devices.

Events

OnConferenced	This event indicates that two calls have been merged into one with no parties removed from the resulting call in process.
OnCallCleared	The Call Cleared event indicates all devices have cleared from the call.
OnConnectionCleared	The Connection Cleared event indicates one device has cleared from the call. For a multiple party call, the call may still exist.
OnDelivered	This event indicates a device is ringing another device or has received an incoming call.
OnDiverted	This event indicates that a call has been diverted from a monitored device. This event is provided for the device from which the call is diverted.
OnEstablished	This event indicates that: a call has been answered, or that a call has been picked up, or that a 3-party conference is now a 2-party call.
OnFailed	This event indicates that a call dialed from a device has failed due to a number of reasons such as invalid digits dialed, the dialed party is busy etc.
OnHeld	This event indicates a device, has been held or is holding another device.

OnNetworkReached	This event indicates a trunk device has been seized for the outgoing call.
OnOriginated	This event indicates that the server is attempting to make a call; it implies that input activity is complete and that a call (rather than feature) has been requested.
OnQueued	This event indicates a monitored device has been queued or a call has been queued at a monitored ACD queue.
OnRetrieved	This event indicates a previously held call has been retrieved and is now connected.
OnServiceInitiated	This event indicates a device has gone offhook manually (i.e., not by a Make Call request).
OnTransferred	This event indicates a device has transferred the held party to the ringing or connected party.

Call event report messages indicate a change in state of one or more connections in the switching sub-domain. Each call event report contains a parameter that summarizes the local connaction state from the perspective of a device that is a monitored object.

Every call event report may contain a cause code to clarify the basic meaning of the event report.

The following event cause codes are common to all events:

- ec_callForwardAlways
- ec_callForwardBusy
- ec_callForwardNoAnswer
- ec_callPickUP
- ec_redirected
- ec_transfer
- ec_callBack
- ec_keyConference

Cause codes specific to an event are listed under each event.

Remarks

None.

See Also

None.

Device.Open Method

Hierarchy

CTNGLib → Device → **Open**

Description

This method opens Device object given by PABX directory number in **dev_id** on existing open PABX given by **pabx** reference. System internally starts monitor on given device. Note that CTNG component library service will not open more then one monitor on the same device id, but will instead pool monitor connections.

Syntax

```
pabx.Open pabx, dev_id
```

Return Value

None.

Parameters

pabx

CTNGLib.PABX, by reference, input, reference to already open PABX object where device can be found.

dev_id

String, by reference, input, directory number of device in PABX.

Remarks

In Visual Basic, the device should be declared with *WithEvents* qualifier, if application developer wants to receive events occurred on device.

In HTML, the device should be inserted in HTML document with <OBJECT> tag referencing CTNGLib.Device **clsid:2BE420EA-C378-40F8-BF73-B19CFF854186** , if application developer wants to receive events occurred on device.

See Also

System.OpenConsole Method

Applies To: System Class

Device.Close Method

Hierarchy

CTNGLib → Device → **Close**

Description

This method opens Device object given by PABX directory number in **dev_id** on existing open PABX given by **pabx** reference. System internally starts monitor on given device. Note that CTNG component library service will not open more then one monitor on the same device id, but will instead pool monitor connections.

Device monitor will not be closed if there are other users having open the same device. When all references to open device are closed, the monitor will be closed after certain amount of time expires and the same device is not reopen in the mean time.

Syntax

```
pabx.Close
```

Return Value

None.

Parameters

None.

Remarks

In Visual Basic, the device should be declared with *WithEvents* qualifier, if application developer wants to receive events occurred on device.

In HTML, the device should be inserted in HTML document with <OBJECT> tag referencing CTNGLib.Device **clsid:2BE420EA-C378-40F8-BF73-B19CFF854186** , if application developer wants to receive events occurred on device.

See Also

System.OpenConsole Method

Applies To: System Class

Switching Function Services

Device.AlternateCall Method

The Alternate Call Service provides the compound action of the Hold Call Service followed by Retrieve Call Service. It places an existing active call on hold and then retrieve a previously held call or connects an alerting call at the same device.

Service Request

The Alternate Call Service Request includes at least one of the following parameters and each parameter has the indicated meaning:

1. CSTA Connection Identifier - indicates the Connected or Alerting Connection to alternate.
2. CSTA Connection Identifier - indicates the Hold Connection to alternate.

The request also may include one or more of the following parameters:

3. CSTA Private Data Information.
4. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

This Service causes the specified device's held and active calls to be swapped. As shown in the figure below, the Alternate Call Service places the user's active call to device D2 on hold and establishes or retrieves the call between device D1 and device D3 as the active call. Device D2 may be considered to be placed automatically on hold immediately prior to the retrieval/establishment of the held/active call to device D3.

Operation of the Alternate Call Service is illustrated in the next figure.

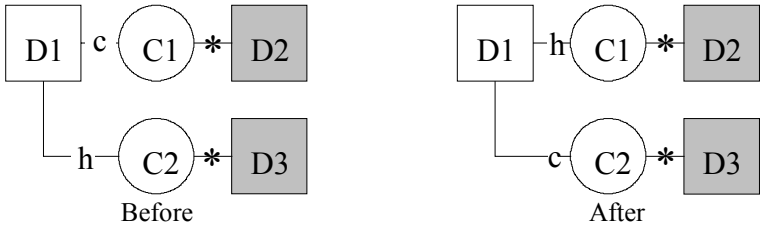

Figure 11 – Alternate Call

Device.AnswerCall Method

Hierarchy

CTNGLib → Device → **AnswerCall**

Description

The Answer Call Service connects an alerting or queued call. The call must be associated with a device that can answer a call without physical manipulation by a user.

Syntax

```
error_status = pabx.Open connection
```

Return Value

CTNGLib.ErrorStatus, the result (positive or negative acknowledgment) of service request.

The positive acknowledgment provides no data.

The negative acknowledgment may return one of the following errors:

- operational error
- state error
- subscribed resource availability error
- system resource error

Parameters

connection

String, by value, input, indicates the Connection ID to answer.

Service Request

The Answer Call Service Request includes the following parameter:

1. CSTA Connection Identifier - indicates the Connection to answer.

The request also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in 9.4, Diagnostic error definitions.

Functional description

The Answer Call Service operates on an incoming call that is alerting or queued at a device. In the following figure, the call, C1 is delivered to device D1. Answer Call is used, typically with telephones that have attached speakerphone units, to establish the call via hands-free operation.

Figure 12 – Answer Call

Remarks

None.

See Also

Applies To: Device Class

Device.AssociateData Method

The Associate Data Service associates CSTA Application Correlator Data, Account Code and/or Authorization Code information with a specified call.

Service Request

The Associate Data Service Request includes the following parameter:

1. CSTA Connection Identifier - indicates the call with which the data is to be associated.

The request also includes at least one of the following parameters:

2. Account Code Data - indicates the account code to associate with the call.
3. Authorization Code Data - indicates the authorization code to allow the call.
4. CSTA Application Correlator Data - contains information supplied by the Computing Function application.

The request also may include one or more of the following parameters:

5. CSTA Private Data Information.
6. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in 9.4, Diagnostic error definitions.

Functional description

The Associate Data Service allows the Computing Function to associate call-specific data with a call. It does not affect the state or progress of the call for which it is invoked.

Device.CallCompletion Method

The Call Completion Service invokes features that complete a call that may otherwise fail or terminate before being answered.

Service Request

The Call Completion Service Request includes the following parameters:

1. CSTA Connection Identifier - indicates the Connection Identifier of the caller.
2. Feature - identifies the feature to invoke. Allowed features are:

   Camp On - queues the call for a device until that device is available.

   Call Back - requests the called device to return the call when the device returns to idle.

   Intrude - adds the call to an existing, active call at the called device.

   Call Back Message - leaves a message for the called user to return the call.

The request also may include one or more of the following parameters:

3. CSTA Private Data Information.
4. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in 9.4, Diagnostic error definitions.

Functional description

Generally, this Service is invoked when a call is set up and encounters a busy called device or no answer.

Camp On allows queuing for availability. Usually, Camp On makes the call wait until the called party finishes a current call and any previously camped on calls.

Call Back requests the called device to return the call when the called device returns to idle. Call Back is similar to Camp On, but with Call Back the call may be hung-up after the Service is invoked. The CSTA Switching Function calls both parties when the called device becomes free.

Intrude allows a call to be added into an existing call at the called device.

Call Back Message leaves a message for the called user to return the call to the caller when the called user is able to do so.

Generic Call Completion Service operation is illustrated in the next figure. In this example, a call, C1, is initiated from device D1 calling device D2, which is busy. The attempt to connect to D2 therefore fails. Callback Call Completion Service (for instance) is then invoked on behalf of D1. When D2 later becomes idle, both D1 and D2 are alerted and when both are answered, a new call, C3, is established between D1 and D2.

Figure 13 – Generic Call Completion

Note that the configuration represented in the “After” diagram above represents the situation after the callback call has been successfully made and answered. Several intermediate, implementation-dependent conditions may exist before the “After” configuration is achieved.

Device.ClearCall Method

The Clear Call Service releases all devices from a specified call and eliminates the call itself. The call ceases to exist and all CSTA identifiers used for observation and manipulation of that call is released.

Service Request

The Clear Call Service Request includes the following parameter:

1. CSTA Connection Identifier - indicates the call to clear.

The request also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

Positive Acknowledgement to the Service Request indicates that all instances of the CSTA Connection Identifiers for all the endpoints in the call and in the current association have become invalid. The Connection Identifiers and their components associated with the just-cleared call can be reused for Services pertaining to other calls. However, these identifiers can not be used to request Services related to the just-cleared call.

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

Each device in the call is released and the CSTA Connection Identifiers (and their components) are freed.

Operation of the Clear Call Service (CSTA Connection Identifier = C1,D1), where call C1 connects devices D1, D2 and D3, is illustrated in the next figure:

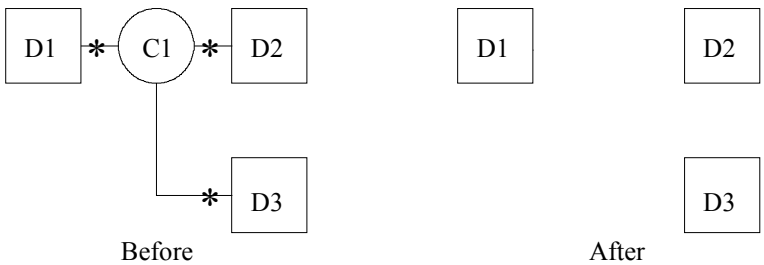

Figure 14 – Clear Call

Device.ClearConnection Method

Hierarchy

CTNGLib → Device → **ClearConnection**

Description

The clear connection service releases a specified device from a designated call. The connection is left in the null state and the connection identifier provided in the service request is automatically released. Note that some connection states (e.g., held) may not allow a connection to be cleared.

Syntax

```
error_status = pabx.ClearConnection connection
```

Return Value

CTNGLib.ErrorStatus, the result (positive or negative acknowledgment) of service request.

The positive acknowledgment provides no data.

The negative acknowledgment may return one of the following errors:

- operational error
- state error
- subscribed resource availability error
- system resource error

Parameters

connection

String, by value, input, indicates the Connection ID to clear.

Remarks

None.

See Also

Applies To: Device Class

Device.ClearConnection Method

The Clear Connection Service releases a specified device from a designated call. The Connection is left in the Null state. Additionally, the CSTA Connection Identifier provided in the Service Request is released.

Service Request

The Clear Connection Service Request includes the following parameter:

1. CSTA Connection Identifier - indicates the Connection to clear.

The request also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

Positive Acknowledgement of the Service Request indicates that the CSTA Connection Identifier for the cleared Connection has been released. The Connection Identifier is not be used to request additional Services from the CSTA server until/unless it is re-allocated by the server.

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

This Service releases the specified Connection and CSTA Connection Identifier from the designated call. The result is as if the device had hung up on the call. Note that the phone might not be physically returned to on-hook and this might result in silence, dial tone, or some other condition. Generally, if only two Connections are in the call, the effect of Clear Connection is the same as that of Clear Call. The following figure illustrates the operation of Clear Connection (CSTA Connection Identifier = C1,D3), where call C1 connects devices D1, D2 and D3. Note that the entire call may not be cleared by this Service if it is some type of conference.

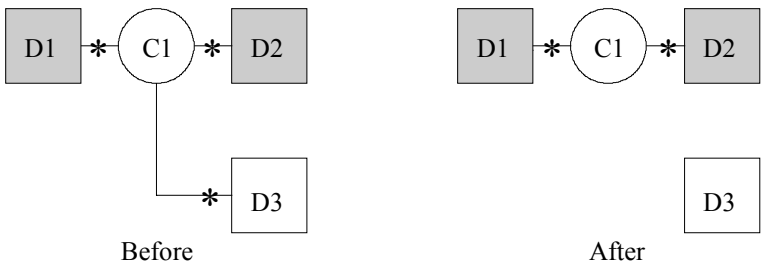

Figure 15 – Clear Connection

Device.ConferenceCall Method

The Conference Call Service creates a conference between an existing held call and another active call at a conferencing device. The two calls are merged into a single call and the two Connections at the conferencing device is resolved into a single Connection in the Connected state. The CSTA Connection Identifiers formerly associated with the conferenced Connections are released, and a new CSTA Connection Identifier for the resulting Connection is created.

Service Request

The Conference Call Service Request includes at least one of the following parameters:

1. CSTA Connection Identifier - indicates the Connection of the held call to be conferenced.
2. CSTA Connection Identifier - indicates the Connection of the active call to be conferenced.

The request also may include one or more of the following parameters:

3. CSTA Private Data Information.
4. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement includes the following parameter:

1. CSTA Connection Identifier - indicates the resulting connection at the conferencing device.

The positive acknowledgement also may provide one or more of the following three parameters for each party that is known to the CSTA sub-domain whose Connection Identifier changes as a result of the Conference Service:

2. CSTA Connection Identifier - indicates the party in the conference.
3. CSTA Device Identifier - provides the Device Identifier for this party in the conference, if that identifier is known.
4. CSTA Connection Identifier - indicates the previous Connection Identifier for this party in its original call.

The positive acknowledgement also may include one or more of the following parameters:

5. CSTA Private Data Information.
6. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The operation of Conference Service is illustrated in the next figure. The starting conditions are that a call C1 from D1 to D2 is in the Hold state and a call C2 from D1 to D3 is in progress.

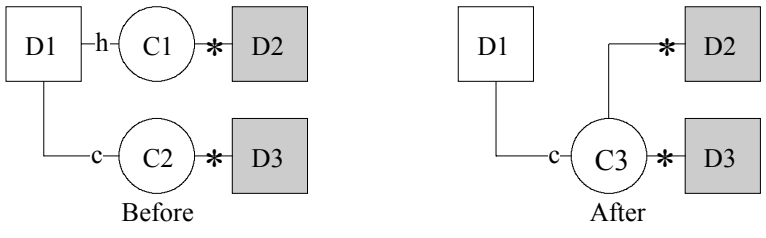

Figure 16 – Conference Call

D1, D2 and D3 are then conferenced together into a single call, C3. The value of the Connection Identifier (D1,C3) may be that of one of the CSTA Connection Identifiers provided in the Conference Service Request (D1,C1 or D1,C2). The Conference Service can be repeated to make n-party conference calls.

Device.ConsultationCall Method

Consultation Call Service provides the compound action of the Hold Call Service followed by Make Call Service. This Service places an existing active call at a device on hold and initiate a new call from the same device.

Service Request

The Consultation Call Service Request includes the following parameters:

1. CSTA Connection Identifier - indicates the existing Connected Connection to be held.
2. CSTA Device Identifier - indicates the device to be consulted.

The request also may include one or more of the following parameters:

3. Device Profile - indicates the Device Profile associated with the device to be consulted.
4. Account Code Data - indicates the account code to associate with the call.
5. Authorization Code Data - indicates the authorization code to allow the call.
6. CSTA Application Correlator Data - contains information supplied by the Computing Function application.
7. CSTA Private Data Information.
8. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement includes the following parameter:

1. CSTA Connection Identifier - indicates the initial Connection to the new call.

The positive acknowledgement also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

Operation of Consultation Call Service is illustrated in the following figure:

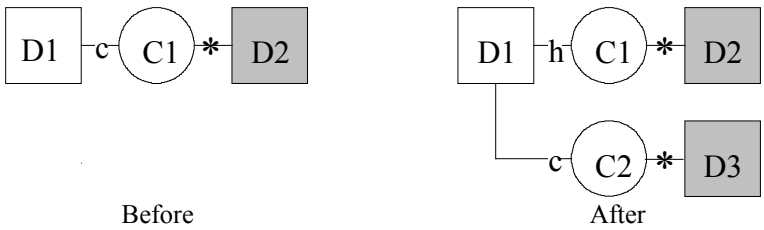

Figure 17 – Consultation Call

Device.DivertCall Method

The Divert Call Service moves a call from one device to another device.

Service Request

The Divert Call Service Request includes the following parameter:

1. Diversion type - indicates the type of diversion requested. This parameter has one of the following values:

   Deflection - means that a Connection is diverted away from a device to a destination that is inside or outside the switching sub-domain.

   Directed pickup - means that a Connection is diverted to a new destination that is inside the switching sub-domain.

   Group pickup - means that a Connection should be diverted to a member of a pickup group.

If the Diversion type parameter specifies Deflection or Directed pickup, then the Service Request includes the following parameter (CSTA Connection Identifier). If the Diversion type parameter specifies Group pickup, then the Service Request does not include the following CSTA Connection Identifier parameter:

2. CSTA Connection Identifier - indicates the call to divert.
3. CSTA Device Identifier - indicates the device to which the call is to be diverted.

The request also may include one or more of the following parameters:

4. Device Profile - specifies the Device Profile of the device to which the call is to be diverted.
5. CSTA Application Correlator Data - contains information supplied by the Computing Function application.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Divert Call Service replaces the originally called device, specified in the CSTA Connection Identifier, with a different called device, specified by a CSTA Device Identifier. The Divert Call Service supports at least three common call diversion functions:

- **Deflection** - Takes a call (typically alerting or queued) at a device and sends it to a new destination. When using Divert Call to perform a deflection, the CSTA Connection Identifier must be included.
- **Pickup** - Takes a call (typically alerting or queued) at another destination and brings it to a device. When using Divert Call to perform a pickup, the CSTA Connection Identifier must be included.
- **Group Pickup** - Takes a call (typically alerting or queued) at one or more predetermined destinations and brings it to a device. When using Divert Call to perform a group pickup, the CSTA Connection Identifier is not included.

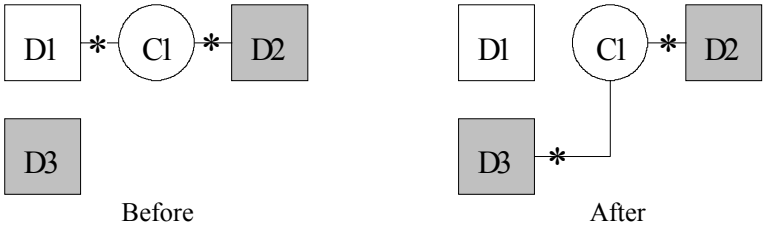

Figure 18 – Divert Call

Device.HoldCall Method

The Hold Call Service places an existing Connection into the Hold state.

Service Request

The Hold Call Service Request includes the following parameter:

1. CSTA Connection Identifier - indicates the Connection to hold.

The request also may include one or more of the following parameters:

2. Connection Reservation - reserves the facility for reuse by the held call. This option is not appropriate for most non-ISDN telephones. The default is no connection reservation.
3. CSTA Private Data Information.
4. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

This Service interrupts communications for an existing call at a device and places that call on hold.

The associated connection is made available for other uses, depending on the reservation option.

Hold Call Service is illustrated in the following figure. Hold Call Service is invoked for device D1 in call C1. Call C1 is then placed on hold. Hold Call maintains the relationship between the holding device and the held call until the call is retrieved from the Hold state or the call is cleared.

Figure 19 – Hold Call

Device.MakeCall Method

Hierarchy

CTNGLib → Device → **MakeCall**

Description

The Make Call Service originates a CSTA call between two devices. The Service creates a new call and establish a Connected Connection with the originating device. The Make Call Service also provides a CSTA Connection Identifier that indicates the Connection of the originating device.

Syntax

```
error_status = pabx.MakeCall device, device2call, diversionOverride
```

Return Value

CTNGLib.ErrorStatus, the result (positive or negative acknowledgment) of service request.

The positive acknowledgment provides no data.

The negative acknowledgment may return one of the following errors:

- operational error
- state error
- subscribed resource availability error
- system resource error

Parameters

device

String, by value, input, indicates the Device ID from which the call originates

Device2call

String, by value, input, indicates the Device ID to which the call should be directed.

diversionOverride

Boolean, by value, input, **MD110 Extension:** if true, the diversion override feature is invoked when making the call.

Service Request

The Make Call Service Request includes the following parameters:

1. CSTA Device Identifier - indicates the device from which the call originates.
2. CSTA Device Identifier - indicates the device to which the call should be directed.

The request also may include one or more of the following parameters:

3. Device Profile - indicates the Device Profile associated with the call.
4. Account Code Data - indicates the account code to associate with the call.
5. Authorization Code Data - indicates the authorization code to allow the call.
6. CSTA Application Correlator Data - contains information supplied by the Computing Function application.
7. CSTA Private Data Information.
8. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement includes the following parameter:

1. CSTA Connection Identifier - indicates the Connection between the originator and the call.

The positive acknowledgement also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

This Service originates a new call from one CSTA device to another. When the Service is initiated the calling device is prompted (if necessary) and when that device acknowledges, a call to the called device is originated.

Make Call operation (Calling device = D1, Called device = D2) is illustrated in the next figure. A call is established as if D1 had called D2 manually. A Connection Identifier for the new Connection (C1,D1) is returned to the client when the allocation condition specified in the Service Request has been satisfied.

Figure 20 – Make Call

After call origination has started, call progress Event Reports selected by the client via the Monitor Start Service may be sent by the server application as connection establishment progresses. The call is not guaranteed to succeed after acknowledgement has been received. For example, there is no assurance that D2 will be answered after it has begun to be alerted or that D2 will not be busy with another call and able to accept the call from D1.

Remarks

Note: ATS and CAS devices will only perform the Make Call service if a call is in service initiated state (i.e., off-hook) on that device.

See Also

Applies To: Device Class

Device.MakePredictiveCall Method

The Make Predictive Call Service originates a CSTA call between two devices. The Service creates a new call and establish a Connection with the called device. The Make Predictive Call Service also provides a CSTA Connection Identifier that indicates the Connection of the called device.

Service Request

The Make Predictive Call Service Request includes the following parameters:

1. CSTA Device Identifier - indicates the device on behalf of which the call is originated.
2. CSTA Device Identifier - indicates the device to which the call should be directed.

The request also may include one or more of the following parameters:

3. Allocation - specifies the condition(s) in which the call attempts to connect to the calling device. If absent, Call Delivered is the default. This parameter has one of the following values:

   Call Delivered - specifies that the call will attempt to connect to the calling device if Alerting or Connected is determined at the called device.

   Call Established - specifies that the call will attempt to connect to the calling device only if Connected is determined at the called device.
4. Device Profile - indicates the Device Profile associated with the call.
5. Account Code Data - indicates the account code to associate with the call.
6. Authorization Code Data - indicates the authorization code to allow the call.
7. CSTA Application Correlator Data - contains information supplied by the Computing Function application.
8. CSTA Private Data Information.
9. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement includes the following parameter:

1. CSTA Connection Identifier - indicates the Connection between the called device and the call.

The positive acknowledgement also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

This Service is often used when calls are made from a group of devices (or a logical device). This Service allocates calls to particular devices within that group at some time during the progress of the call.

The Service first initiates a call to the called device. Depending on the call's progress, the call may be connected with an "originating" device during the progress of the call. The point at which the call attempts to connect to the originating device is determined by the allocation parameter. If the allocation parameter is set to Call Delivered, then the call is allocated upon detection of an Alerting or Connected state at the called device. If the allocation parameter is set to Call Established, the call is allocated upon detection of a Connected state at the called device.

The typical use of this Service is to place calls out of a CSTA sub-domain. The CSTA Connection Identifier provided by this Service may apply to an outbound trunk (rather than the terminating, called device).

The result of Make Predictive Call (Calling device = group device D1, Called device = D2) is illustrated in the following figure:

Figure 21 – Make Predictive Call

Device.ParkCall Method

The Park Call Service moves a Connected call at one device to another (parked-to) device. The device on whose behalf Park Call was invoked (the Parking device) is no longer associated with the call.

Service Request

The Park Call Service Request includes the following parameters:

1. CSTA Connection Identifier - indicates the existing Connected Connection of the Parking device.
2. CSTA Device Identifier - indicates the device (i.e. the Parked-to device) at which the call should be parked.

The request also may include one or more of the following parameters:

3. Device Profile - specifies the Device Profile for the "Parked-to" device.
4. CSTA Application Correlator Data - contains information supplied by the Computing Function application.
5. CSTA Private Data Information.
6. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement also may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

As shown in the following figure, when Park Call Service is invoked for device D1 in call C1, the call C1 becomes queued at device D3. Device D1 is no longer associated with call C1.

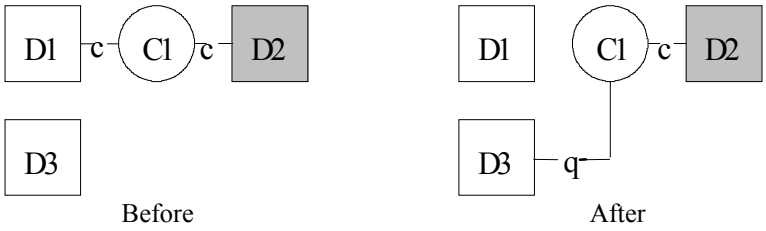

Figure 22 – Park Call

Device.QueryDevice Method

The Query Device Service provides indication of the state of device features or static attributes.

Service Request

The Query Device Service Request includes the following parameters:

1. CSTA Device Identifier - indicates the device to query.
2. Feature - indicates the requested information and consists of exactly one of the following:
   a. **Message Waiting** - indicates available waiting messages.
   b. **Do Not Disturb** - indicates whether the device is in the Do Not Disturb state.
   c. **Forward** - indicates whether the device is forwarding calls and, if so, what type of forwarding is in place and the number forwarded to.
   d. **Device Information** - indicates the class and type of device and, optionally, a Short Form Device Identifier for the device.
   e. **Agent State** - indicates ACD agent state.
   f. **Routing Enabled** - indicates that the device may make route requests of the Computing Function.
   g. **Auto Answer** - indicates the auto-answer status of the device.
   h. **Microphone Mute** - indicates whether the device's microphone is Off (i.e. muted) or On.
   i. **Speaker Mute** - indicates whether the device's speaker is Off (i.e. muted) or On.
   j. **Speaker Volume** - indicates the current volume setting of the device.

The request also may include one or more of the following parameters:

3. CSTA Private Data Information.
4. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement includes the following parameter:

1. Feature - provides the requested information and consists of the same lettered item as was indicated in the Service Request:
   a. On/Off - indicates available waiting messages.
   b. On/Off - indicates whether the device is in the Do Not Disturb state.
   c. Type of Forwarding - indicates each of the following that are on, and may return others of these that are off as well:

      Immediate - Forwarding all calls. If provided then the response also includes:

      On/Off - indicates whether the device is forwarding calls.

      Device Identifier - indicates the device to which calls are forwarded.

      Busy - Forwarding when busy. If provided then the response also includes:

      On/Off - indicates whether the device is forwarding calls.

      Device Identifier - indicates the device to which calls are forwarded.

      No Answer - Forwarding after no answer. If provided then the response also includes:

      On/Off - indicates whether the device is forwarding calls.

      Device Identifier - indicates the device to which calls are forwarded.

Immediate Internal - Forwarding all internal calls. If provided then the response also includes:

On/Off - indicates whether the device is forwarding calls.

Device Identifier - indicates the device to which calls are forwarded.

Immediate External - Forwarding all external calls. If provided then the response also includes:

On/Off - indicates whether the device is forwarding calls.

Device Identifier - indicates the device to which calls are forwarded.

Busy Internal - Forwarding when busy for an internal call. If provided then the response also includes:

On/Off - indicates whether the device is forwarding calls.

Device Identifier - indicates the device to which calls are forwarded.

Busy External - Forwarding when busy for an external call. If provided then the response also includes:

On/Off - indicates whether the device is forwarding calls.

Device Identifier - indicates the device to which calls are forwarded.

No Answer Internal - Forwarding after no answer for an internal call. If provided then the response also includes:

On/Off - indicates whether the device is forwarding calls.

Device Identifier - indicates the device to which calls are forwarded.

No Answer External - Forwarding after no answer for an external call. If provided then the response also includes:

On/Off - indicates whether the device is forwarding calls.

Device Identifier - indicates the device to which calls are forwarded.

d. Device Information - indicates device class and type. It also may indicate a Short Form Device Identifier for the device. The class specification information includes one or more of the following attributes: Voice, Data, Image, Audio, Other.

   The type specification information includes one or more of the following: station, line, button, ACD, trunk, operator, other, station group, line group, button group, ACD group, trunk group, operator group, other group, conference bridge, park device.

e. Agent State - indicates ACD agent state. The state is one of the following values (which have been previously defined in conjunction with the Agent State model in Figure 8):

   Agent Null

   Agent Not Ready

   Agent Ready

   Agent Busy

   Agent Working After Call

f. Routing Enabled - indicates whether the device may make route requests of the Computing Function.

g. On/Off - indicates whether Auto-answer is On or Off.

h. On/Off - indicates whether the device's microphone is On or Off (i.e. muted).

i. On/Off - indicates whether the device's speaker is On or Off (i.e. muted).

j. Speaker Volume - indicates the current volume setting of the device. The value is an integer in the range 0 to 100.

The positive acknowledgement also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Device.ReconnectCall Method

The Reconnect Call Service provides the compound action of the Clear Connection Service followed by the Retrieve Call Service. It clears an existing Connection and then retrieve a previously held Connection at the same device.

Service Request

The Reconnect Call Service Request includes at least one of the following parameters:

1. CSTA Connection Identifier - indicates the Connection to be cleared.
2. CSTA Connection Identifier - indicates the Connection to be retrieved.

The Service Request also may include one or more of the following parameters:

3. CSTA Private Data Information.
4. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

Reconnect Call Service causes an existing, Connected Connection to be dropped. Having dropped the call, the specified held call is retrieved and becomes active. This Service is commonly used to drop an active call and return to a held call. However, it can also be used to cancel a consultation call (because of no answer, called device busy, etc.) and then return to a held call.

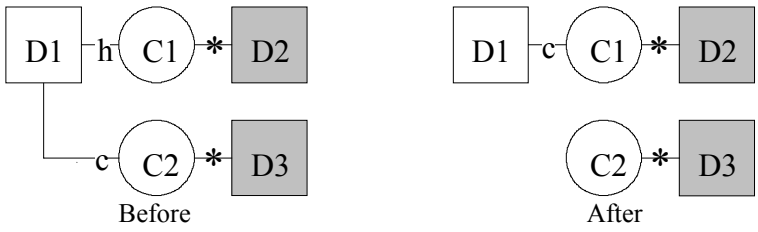

Figure 23 – Reconnect Call

Device.RetrieveCall Method

The Retrieve Call Service connects an existing Hold Connection.

Service Request

The Retrieve Call Service Request includes the following parameter:

1. CSTA Connection Identifier - indicates the Connection to be retrieved.

The request also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The indicated Connection is restored to the Connected state. The call's state may change due to actions by far-end endpoints. If the Hold Call Service reserved the Hold Connection and the Retrieve Call Service is requested for the same call, then the Retrieve Call Service uses the reserved Connection.

Figure 24 – Retrieve Call

Device.SendDTMFTones Method

The Send DTMF Tones Service places DTMF Tones on behalf of a Connection in a call.

Service Request

The request for Send DTMF Tones Service includes the following parameters:

1. CSTA Connection Identifier - indicates the Connection on which the DTMF signals are to be sent.
2. Character String to Send - indicates a string of characters to be sent, encoded in DTMF format, by the Switching Function on a specified connection. All characters from the set {1234567890#*} are encoded as DTMF tones. Additionally, characters from the set {ABCD} may be encoded as DTMF tones, and the request string may contain other additional characters as well. Handling of any of the additional characters is Switching Function dependent.

The request also may include one or more of the following parameters:

3. Tone Duration - indicates the duration, in milliseconds, of the tones to be provided.
4. Pause Duration - indicates the duration, in milliseconds, of the silences between tones to be provided.
5. CSTA Private Data Information.
6. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

This Service causes a series of DTMF tones, corresponding to (certain) IA5 characters in a specified string, to be sent by the switch on a specified call. Only characters that can be encoded as DTMF tones can be sent as DTMF tones by the Switching Function. However, other valid characters contained in the Characters to Send parameter are either ignored or may be used by the Switching Function to control the sending of DTMF tones. For example, a “,” character may cause the Switching Function to insert a pre-defined pause between successive DTMF tones. However, any such significance attached to characters in the supplementary set of characters is entirely switch-dependent and not covered by this Service definition.

This Service definition also supports optional parameters to control tone cadence. If these are not present or cannot be met, the switch may use its default cadences associated with each device over which the tones are sent.

Device.SetFeature Method

The Set Feature Service sets features at a device. It does not set system features or allow administration.

Service Request

The Set Feature Service Request includes the following parameters:

1. Device Identifier - indicates the device on which to set the feature.
2. Feature - indicates the requested feature to set. This parameter has one of the following values:
   a. **Message Waiting** - sets messages available. If this parameter is chosen, the following additional parameter is included:
      3. On/Off - indicates whether to turn on or off.
   b. **Do Not Disturb** - sets Do Not Disturb. If this parameter is chosen, the following additional parameter is included:
      4. On/Off - indicates whether to turn on or off.
   c. **Forward** - sets forwarding calls. If this parameter is chosen, the following additional parameters (5. and 6.) is included:
      5. Type of Forwarding - the value is one of the following with meaning as indicated:

         Immediate - Forwarding all calls,

         Busy - Forwarding when busy,

         No Answer - Forwarding after no answer,

         Immediate Internal - Forwarding all internal calls,

         Immediate External - Forwarding all external calls,

         Busy Internal - Forwarding when busy for an internal call,

         Busy External - Forwarding when busy for an external call,

         No Answer Internal - Forwarding after no answer for an internal call,

         No Answer External - Forwarding after no answer for an external call.

         For each forwarding type, the value indicates whether to turn forwarding “on” or “off”. If ”on” then the following parameter may be included:
      6. Device Identifier - indicates the device to which calls are forwarded.
   d. **Agent Parameter** - the ACD Agent State results from this parameter. This parameter has one of the following values:

      Agent Working After Call.

      Agent Ready.

      Agent Logged On. If this value is specified, then the request also may include one or more of the following parameters:

      7. String - specifies a password or authorization code.
      8. Agent Identifier - specifies the Agent Identifier.
      9. Device Identifier - specifies the ACD pilot or group into which the agent is logging on.

      Agent Logged Off. If this value is specified, then the request also may include one or more of the following parameters:

      10. String - specifies a password or authorization code.
      11. Agent Identifier - specifies the Agent Identifier.
      12. Device Identifier - specifies the ACD pilot or group from which the agent is logging out.

      Agent Not Ready.

e. **Enable Routing** - indicates whether route requests should be made from the device. If this value is chosen, then the following additional parameter is included:

13. This parameter indicates routeing enabled or disabled.

Note 4:

Route requests may be made by the Switching Function without the involvement of a particular device. Additionally, the device may request routes with a call in any state - including Null.

f. **Auto Answer** - sets whether a call arriving at a device should be answered automatically. If this value parameter is chosen, then the following additional parameter is included:

14. This parameter indicates enabled or disabled.

g. **Microphone Mute** - sets the device's microphone to Off (i.e. muted) or On. If this parameter is chosen, the following additional parameter is included:

15. On/Off - indicates whether to set the microphone on or off.

h. **Speaker Mute** - sets the device's speaker to Off (i.e. muted) or On. If this parameter is chosen, the following additional parameter is included:

16. On/Off - indicates whether to set the speaker on or off.

i. **Speaker Volume** - sets the device's volume level.

17. Speaker Volume - indicates the volume level to set. Values may range from 0 to 100, with a value of 0 indicating silence and 100 indicating maximum volume. The granularity and accoustic loudness effect of this parameter is switch- and device-specific.

The request also may include one or more of the following parameters:

18. Device Profile - indicates the Device Profile associated device / feature to set.

19. CSTA Private Data Information.

20. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Device.SingleStepConference Method

The Single Step Conference Service joins a device into an existing call. This Service creates a Connected Connection between an existing call and a device.

Service Request

The Single Step Conference Service Request includes the following parameters:

1. CSTA Connection Identifier - indicates an existing Connection in the call into which a new device will be added.
2. CSTA Device Identifier - indicates the device to be added into the call.

The Service Request also may include one or more of the following parameters:

3. Type of Operation - indicates Silent or Active participation of the conferenced device.
4. Device Profile - indicates the Device Profile associated with the added device.
5. Account Code Data - indicates the account code to associate with the resulting call.
6. Authorization Code Data - indicates the authorization code to allow the call.
7. CSTA Application Correlator Data - contains information supplied by the Computing Function application.
8. CSTA Private Data Information.
9. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement includes the following parameter:

1. CSTA Connection Identifier - indicates the Connection of the device that was added to the call.

The positive acknowledgement also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

Operation of the Single Step Conference Service is illustrated in the following figure. Devices D1 and D2 are both involved (in any state except NULL) with call C1. When Single Step Conference is invoked, device D3 is conferenced into call C1. Call C1 retains its identity in that a new Call Identifier is not created. However, a new Connection Identifier (D3,C1) is created. This operation differs from that of Conference Service in that call C1 need not be on hold at either D1 or D2 (although it is allowed to be on hold at either or both) before the conferencing action takes place. The Single Step Conference Service can be repeated to make n-party conference calls.

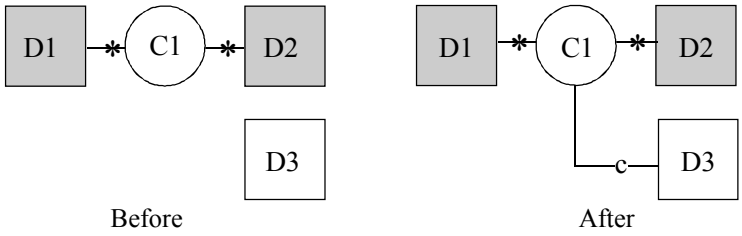

Figure 25 – Single Step Conference

Device.SingleStepTransferCall Method

The Single Step Transfer Call Service replaces a device with a Connected Connection with a new device that is not currently participating in the call. The device that has been replaced is no longer involved with the call.

Service Request

The Single Step Transfer Call Service Request includes the following parameters:

1. CSTA Connection Identifier - indicates the existing Connected Connection to be replaced.
2. CSTA Device Identifier - indicates the new called (transferred-to) device.

The request also may include one or more of the following parameters:

3. Device Profile - indicates the Device Profile associated with the new called (transferred-to) device.
4. Account Code Data - indicates the account code to associate with the call.
5. Authorization Code Data - indicates the authorization code to allow the call.
6. CSTA Application Correlator Data - contains information supplied by the Computing Function application.
7. CSTA Private Data Information.
8. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include the following parameter:

1. CSTA Connection Identifier - indicates a resulting Connection in the remaining call.

The positive acknowledgement also may include one or more of the following three parameters for each party that is known to the CSTA sub-domain whose Connection Identifier changes as a result of the Single Step Transfer Call Service:

2. CSTA Connection Identifier - indicates the party in the resulting call.
3. CSTA Device Identifier - provides, if known, the static reference for the party in the resulting call.
4. CSTA Connection Identifier - indicates the previous identifier for the party in its original call.

The positive acknowledgement also may include one or more of the following parameters:

5. CSTA Private Data Information.
6. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

Operation of the Single Step Transfer Call Service is illustrated in the next figure. The starting conditions are: a call C1 between D1 and D2 is in the Established state. This Service, if issued on behalf of D1, transfers the existing call, C1, to create a new call from device D2 to device D3. Device D1 is released from the call. Single Step Transfer differs from Transfer in that Transfer requires that a call explicitly be placed on hold at a device before it is transferred from that device.

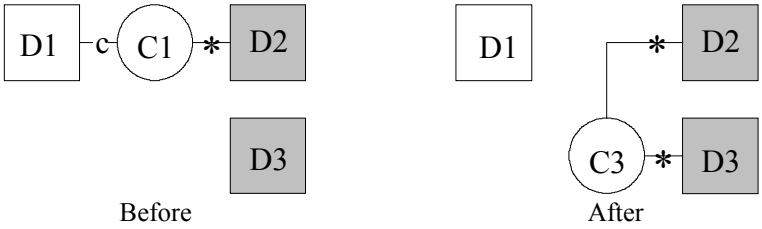

Figure 26 – Single Step Transfer Call

Device.TransferCall Method

The Transfer Call Service transfers a call held at a device to an active call at the same device. The held and active calls at a common device is merged into a new call. Also, the Connections of the held and active calls at the common device becomes Null and their CSTA Connections Identifiers are released.

Service Request

The Transfer Call Service Request includes at least one of the following parameters:

1. CSTA Connection Identifier - indicates the Hold Connection to transfer.
2. CSTA Connection Identifier - indicates the Connected Connection to which the held call should be transferred.

The request also may include one or more of the following parameters:

3. CSTA Private Data Information.
4. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include the following parameter:

1. CSTA Connection Identifier - indicates a resulting Connection in the remaining call.

The positive acknowledgement may include one or more of the following three parameters for each party that is known to the CSTA sub-domain whose Connection Identifier changes as a result of the Transfer Call Service:

2. CSTA Connection Identifier - indicates the party in the resulting call.
3. CSTA Device Identifier - provides, if known, the static reference for the party in the resulting call.
4. CSTA Connection Identifier - indicates the previous Connection Identifier for the party in its original call.

The positive acknowledgement also may include one or more of the following parameters:

5. CSTA Private Data Information.
6. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

Operation of the Transfer Call Service is illustrated in the next figure. The starting conditions are: a call, C1, from D1 to D2 is in held state. A call, C2, from D1 to D3 is in progress. This Service transfers the held call, C1, between devices D1 and D2 into a call from device D2 to device D3.

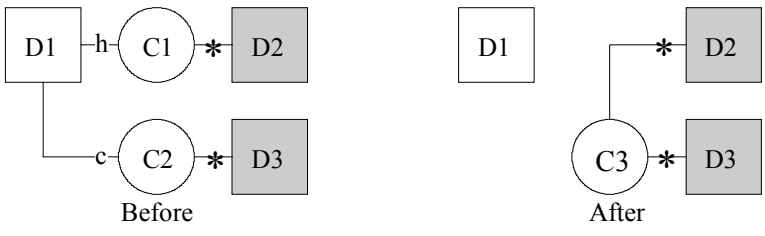

Figure 27 – Transfer Call

The Transfer Call Service is used when a call, C2, from D1 to D3, is in any state other than the Failed or Null state. When Transfer Call Service successfully completes, D1 is released from the call.

Bi-directional Services

System.Escape Method

The Escape Service allows an implementation, using CSTA protocol and defined CSTA Services, to provide services that are not defined by CSTA Standard. The Escape Service uses Object Identifiers, as described in ASN.1 (see CCITT Recommendations X.208/X.209, ISO International Standards ISO/IEC 8824 / ISO/IEC 8825).

Service Request

The Escape Service Request contains the following parameter:

1. CSTA Private Data Information.

The Service Request also may include the following parameter:

2. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

While most common switching and computing Services required by CSTA are standardized, there is a need to be able to “escape” from standard operations in order to exploit some special feature of a manufacturer’s switch or computer. A mechanism is also needed to allow manufacturers to experiment with new services that may, at a later date, become standardized.

If the server can perform the requested service it will do so using the Private Data provided.

System.SystemStatus Method

The System Status Service reports the status of the switching or computing system.

Service Request

The System Status Service Request contains the following parameter:

1. Cause - indicates the reason for the System Status Service request. The cause is one or more of the following:

   Disabled - indicates that existing Monitor Requests have been disabled. Other Requests and Responses also may be disabled, but reject responses are always provided.

   Enabled - indicates that Requests and Responses have been enabled. This usually occurs after a disruption or restart. This status cause is always sent after an Initializing cause has been sent and may be sent under other conditions. This status indicates that there are no outstanding monitor requests.

   Initializing - indicates that the system is initializing or restarting. This status indicates that a system is temporarily unable to respond to any requests. If provided, this status message is followed by an Enable status message to indicate that the Init process has completed.

   Messages Lost - indicates that Requests and/or Responses, including Event Reports, may have been lost.

   Normal - may be sent at any time and indicates that the status is normal. This status has no effect on other Services.

   Overload Imminent - indicates that the receiver is requested to take initiative to shed load.

   Overload Reached - indicates that the requester may take initiative to shed load. This cause may be followed by Stop Monitor requests sent to the client and by rejections to additional Service Requests.

   Overload Relieved - indicates that the overload condition has passed.

The Service Request also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The System Status Service performs no action other than informing.

Status Reporting Services

Device.SnapshotCall Method

Snapshot Call Service provides information about a specified CSTA call. The information provided includes the identities of the devices and their Connections in the call as well as the states of those Connections that comprise an overall Call state.

Service Request

The Snapshot Call Service Request includes the following parameter:

1. CSTA Connection Identifier - indicates the call to be snapshot.

The request also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement includes the following parameters for every endpoint in the call:

1. CSTA Device Identifier - indicates the CSTA Device Identifier for the device. If the device is outside the switching sub-domain, then the parameter may indicate “Unknown”.
2. CSTA Connection Identifier - indicates the CSTA Connection Identifier for the endpoint.

The positive acknowledgement also includes the following parameter for each endpoint within the CSTA switching sub-domain that is in the call:

3. CSTA Connection state - indicates the Connection state for the endpoint. This state is one of the following: Null, Initiated, Alerting, Connected, Hold, Failed, Queued.

The positive acknowledgement also may include the following parameter for one or more endpoints in the call:

4. Device Profile - indicates the Device Profile associated with the device.

The positive acknowledgement also may include one or more of the following parameters:

5. CSTA Private Data Information.
6. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgment includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Snapshot Call Service is intended to provide information about calls that makes further monitoring more meaningful. For example, if a CSTA application were to start working with a call, the Event Reports needed to provide synchronization may not occur for some time. To facilitate operations before an Event Report is available to synchronize the monitor, it is necessary to be able to query the current state of CSTA objects. Snapshot Call Service provides that function.

Snapshot Call Service obtains current call status and returns it in a response. This does not affect the states of any objects in the Switching Function.

Device.SnapshotDevice Method

Snapshot Device Service provides information about calls associated with a given CSTA device. The information provided is sufficient to identify each call and indicate each call's local connection state.

Service Request

The Snapshot Device Service Request includes the following parameter:

1. CSTA Device Identifier - indicates the device to be snapshot.

The request also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement includes the following parameters for each call at the device and also may include the Device Profile parameter for each call at the device:

1. CSTA Connection Identifier - indicates the call and, for some implementations, the device's dynamically-assigned Device Identifier for the call.
2. CSTA Connection States - is a list of the states of the Connections of the call identified in parameter 1 and the snapshot device. This list provides either the local Connection state or an overall call state. The call state may be provided as a sequence of Connection states unless that sequence is the equivalent of a CSTA defined call state. If the call state is one of the CSTA defined states, then the entire state is provided as a composite call state rather than a sequence of individual states. Sub-clause 6.1.5, Call states, contains additional information on Call states.
3. Device Profile - indicates the Device Profile associated with the call.

The positive acknowledgement also may include one or more of the following parameters:

4. CSTA Private Data Information.
5. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Snapshot Device Service is intended to provide information about devices to make further monitoring more meaningful. For example, when a CSTA application starts working with a device, the Event Reports needed to provide synchronization may not occur for some time. To facilitate operations before Event Reports synchronize the monitor, it is necessary to be able to query the current state of CSTA objects. Snapshot Device Service provides that function.

Snapshot Device Service obtains current device status and returns it in a response. This does not affect the states of any objects in the Switching Function.

Computing Function Services

Device.ReRoute Method

The Re-Route Service requests an alternate destination from the one provided by a previous Route Select Service and based on previous information provided for the call.

Service Request

The Re-Route Service Request includes the following parameter:

1. CSTA Cross Reference Identifier - indicates the set of routeing Services that are used for a particular call.

The Re-Route Service Request also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Service Response

The server does not give positive acknowledgement to the Re-Route Service Request. However, the server may provide a negative acknowledgement of the Service Request. The requested route is provided via a Route Select Service message sent from the server to the client.

Positive acknowledgement

This Service does not have a positive acknowledgement.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The requested route is sent from the Computing Function to the Switching Function by the Route Select Service. The client is expected to use the Cross Reference Identifier generated by the initial Route Request Service to link this Service to the others that are used to provide a route.

Device.RouteEnd Method

The Route End Service ends a routeing dialogue. It may be invoked by the client or server.

Service Request

The Route End Service Request includes the following parameter:

1. CSTA Cross Reference Identifier - indicates the set of routeing Services that are used for a particular call. This CSTA Cross Reference Identifier instance becomes invalid.

The request also may include one or more of the following parameters:

2. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.
3. CSTA Private Data Information.
4. CSTA Security Service Information.

The server verifies that the Service Request is correct and may notify the client application in order to acknowledge or reject the Service Request.

Service Response

The server does not give positive acknowledgement to the Route End Service Request. However, the server may provide negative acknowledgement of the Service Request.

Positive acknowledgement

This Service does not provide a positive acknowledgement.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Route End Service may be invoked by the Switching Function when a call has been successfully routed, cleared, or when the Computing Function has failed to provide a route within a time limit. It can also be provided by the Computing Function to indicate that no (more) routes are (currently) available for the requested destination.

Device.RouteRequest Method

The Route Request Service requests a destination for a call. To aid in the selection of a destination, the Service Request includes the current destination and may include additional information.

Service Request

The Route Request Service Request includes the following parameters:

1. CSTA Cross Reference Identifier - indicates the set of routeing Services that are used for a particular call.
2. Current Route - indicates the current destination of the call for which a route is requested.

The Route Request Service Request also may include one or more of the following parameters:

3. Calling Device - indicates the originator of the call.
4. Routing Device - indicates the device which initiated the Route Request Service.

   Note 5:

   Route requests may be made by the Switching Function without the involvement of a particular device. Additionally, the device may request routes with a call in any state - including Null.

5. CSTA Connection Identifier - indicates the CSTA Call Identifier.
6. Route Selection Algorithm - indicates the type of routeing algorithm requested. This parameter may have the following values:

   ACD - indicates that the route should be selected using an algorithm that distributes calls to multiple devices.

   Emergency - indicates that the call is emergency and a suitable route should be selected.

   Least Cost - indicates that a route costing the least among the available routes should be provided.

   Normal - indicates that a default route is requested.

   User Defined - indicates an application-defined routeing algorithm.

7. Priority - indicates the call's priority. This may affect selection of alternative routes.
8. Device Profile - contains an ISDN Device Profile for the calling device.
9. CSTA Application Correlator Data - provides application-specific data associated with the call.
10. CSTA Private Data Information.
11. CSTA Security Service Information.

Service Response

The server does not give positive acknowledgement to the Service Request. However, the server may provide a negative acknowledgement of the Service Request. The requested route is provided via a Route Select Service sent from the server to the client.

Positive acknowledgement

This Service does not provide a positive acknowledgement.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

If the information in the request is invalid, the Service Request may be negatively acknowledged. The requested route is sent from the Computing Function to the Switching Function by the Route Select Service. The client is expected to generate the Cross Reference Identifier to link this Service to the others that are used to provide a route.

Device.RouteSelect Method

The Route Select Service provides the client with a destination requested by a previous Route Request or Re-Route Service.

Service Request

The Route Select Service Request includes the following parameters:

1. CSTA Cross Reference Identifier - indicates the set of routeing Services that are used for a particular call.
2. Route Selected - indicates the selected destination of the call for which a route was requested.

The request for the Route Select Service also may include one or more of the following parameters:

3. Device Profile - indicates the Device Profile of the called destination.
4. Remaining Retries - indicates the number of alternative routes remaining. This element may have a special value that indicates that the server does not keep count, or that there is no fixed list.
5. Route Used Request - indicates a request to receive a Route Used Service after providing the route.
6. CSTA Application Correlator Data - contains information supplied by the application.
7. CSTA Private Data Information.
8. CSTA Security Service Information.

Service Response

The server does not provide positive acknowledgement of the Service Request. However, the server may provide a negative acknowledgement. The Route Select Service is completed via a Route End Service initiated by either the server or client.

Positive acknowledgement

This Service does not provide a positive acknowledgement.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The requested destination is provided by the Route Select Service. The client is expected to use the Cross Reference Identifier generated by the original Route Request Service to link this Service to the others that are used to provide a route.

Device.RouteUsed Method

The Route Used Service provides the actual destination for a call that has been routed using the Route Select Service with its optional parameter that requests the route that was used.

Service Request

The Route Used Service Request includes the following parameters:

1. CSTA Cross Reference Identifier - indicates the set of routeing Services that are used for a particular call.
2. Route Used - indicates the selected destination of the call for which a route was requested.

The request also may include one or more of the following parameters:

3. Calling Device - indicates the originator of the call.
4. Domain - indicates whether the resolved endpoint is within the CSTA switching sub-domain or whether the call has been routed outside the CSTA switching sub-domain.
5. CSTA Application Correlator Data - contains information previously supplied by the application.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Service Response

The server does not provide positive acknowledgement of the Service Request. However, the server may provide a negative acknowledgement. The Route Used Service is completed via a Route End Service sent by either the server or client.

Positive acknowledgement

This Service does not provide a positive acknowledgement.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Route Used Service can be used to inform the server of the route that the client selected. Often the route returned by the server is altered by Forwarding or Do Not Disturb features, or is resolved by an ACD from the pilot to a particular agent.

Voice Unit Services

The Voice Unit is an example (subclass) of a Special Resource Function (SRF) introduced in the modeling clauses of CSTA Standard. These Services have been developed specifically for voice, but it is recognized that they could be used for other media, such as fax or video. The model represented in Figure 9 illustrates the relationships among the following Services. The interface to the Voice Unit provides the following Services to manipulate voice messages stored in the Voice Unit.

Voice.ConcatenateMessage Method

Concatenate Message combines multiple messages, in the sequence provided, into a single resulting message. The messages that form the concatenated message are not deleted or otherwise changed. This Service provides a unique Message Identifier for the concatenated message that remains valid until the concatenated message is deleted.

Service Request

The Concatenate Message Service Request contains two or more instances of the following parameter:

1. Message Identifier - indicates message to be concatenated.

The Service Request also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement includes the following parameter:

1. Message Identifier - indicates the resulting concatenated message.

The positive acknowledgement may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Concatenate Message Service causes the Voice Unit to concatenate specified messages into a single new message, the Message Identifier of which is returned in the Service Response. The original messages are preserved by this function. If it is not desired to retain the messages that are used to create the new message, the Delete Message Service can be used.

The Message Identifier returned by this Service is valid until the new message formed by the concatenation is deleted.

Voice.DeleteMessage Method

Delete message deletes a specified voice message.

Service Request

The Delete Message Service Request contains the following parameter:

1. Message Identifier - indicates the message to be deleted.

The Service Request also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Delete Message Service causes the Voice Unit to delete a specified message.

Voice.PlayMessage Method

The Play Message Service causes the Voice Unit to start playing a voice message on a particular Connection. Play Message may play a second message on a Connection that has a currently-suspended message. However, it does not play the same message over the same Connection. If the Duration parameter is set to zero, then the message will go immediately to the Suspend Play State.

Service Request

The Play Message Service Request contains the following parameters:

1. Message Identifier - indicates the message to be played.
2. CSTA Connection Identifier - indicates the Connection on which the message is to be played.

The Service Request also may include one or more of the following parameters:

3. Duration - time of message to play in milliseconds.
4. Termination Conditions - mask of actions that will cause playback to terminate.
5. CSTA Private Data Information.
6. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Play Message Service causes the Voice Unit to start playing a designated message until the reception of a valid Stop Service Request or Suspend Service Request or until the end of the message is detected. Not allowing the same Message Identifier twice ensures that the Message Identifier remains unique for all messages associated with a particular voice path / Connection Identifier. Note that media conversion may be provided in this Service.

Voice.QueryVoiceAttribute Method

The Query Voice Attribute Service accesses and report current value of a specified voice attribute for a specified Message.

Service Request

The Query Voice Attribute Service Request contains the following parameters:

1. Message Identifier - indicates the message whose attribute should be queried.
2. Attribute - indicates the attribute to be queried. This parameter has one of the following values if parameter 3 (Connection Identifier) is not provided:

   Encoding algorithm - requests the encoding algorithm used for the specified message.

   Sampling Rate - requests the sampling rate used for the specified message.

   Duration - requests the duration of the specified message.

   File Name - requests the file name of the specified message.

   If parameter 3 (Connection Identifier) is provided, then parameter 2 (Attribute) has any one of the four above-listed values or any one of the following five values:

   Current Position - requests the current position in the message. If requested, this attribute will be reported in milliseconds from the beginning of the specified message.

   Current Speed - requests the current playing speed of the specified message. If requested, this reported value represents percentage of normal speed, with 100% being normal speed and the slowest reportable speed being 1%.

   Current Volume - requests the current playing volume of the specified message. If requested, the reported value is an integer from 0 to 100, with 100 indicating maximum volume and 0 indicating minimum volume.

   Current Level - requests the current recording level of the specified message. If requested, the reported value is an integer from 0 to 100, with 100 indicating maximum level and 0 indicating minimum level.

   Current State - requests the current state of the specified message. The reported value indicates one of the following states:

   Stop
   Play
   Record
   Suspend Play
   Suspend Record
   Review

The Service Request also may include one or more of the following parameters:

3. CSTA Connection Identifier - indicates the Connection of the message whose attribute is being queried.
4. CSTA Private Data Information.
5. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement includes the following parameter:

1. Attribute - indicates the value of the requested attribute.

The positive acknowledgement also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Query Voice Attribute Service gets the value of a specified voice attribute of the current message.

Voice.RecordMessage Method

The Record Message Service starts recording a voice message from a specified Connection.

Service Request

The Record Message Service Request contains the following parameter:

1. CSTA Connection Identifier - indicates the Connection from which the message is to be recorded.

The Service Request also may include one or more of the following parameters:

2. Sampling Rate - specifies the sampling rate to be used when recording.
3. Encoding Algorithm - specifies the encoding algorithm to be used during recording.
4. Maximum Duration - specifies, in milliseconds, the maximum message time to record.
5. Termination Conditions - specifies, via a mask, the actions that will cause recording to terminate.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement includes the following parameter:

1. Message Identifier - indicates the message to be recorded.

The positive acknowledgement also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Record Message Service provides a Message Identifier in the positive acknowledgement and causes the Voice Unit to begin recording from a specified Connection. If the sampling rate or encoding algorithm are not specified or cannot be provided, then the Voice Unit may use its default sampling rate and/or encoding algorithm.

Voice.Reposition Method

The Reposition Service moves the current position pointer forward or backward a specified number of milliseconds in a message. It also allows setting the position pointer to the start or to the end of the message.

Service Request

The Service Request contains the following parameters:

1. CSTA Connection Identifier - indicates the Connection on which the message is present.
2. Period - indicates the number of milliseconds to move forward or backward. A positive period moves forward (toward the end of the message); a negative period moves backward. It also may have a value indicating "Beginning of Message" and a value indicating "End of Message".

The Service Request also may include one or more of the following parameters:

3. Message Identifier - indicates the message to reposition.
4. CSTA Private Data Information.
5. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Reposition Service causes the Voice Unit to reposition the message pointer according to the number of milliseconds indicated in the period of the request. Moving forward and backward within the indicated message on the indicated Connection is accomplished by using, respectively, positive or negative values for the period.

Voice.Resume Method

The Resume Service restarts the playing or recording of a previously suspended message at the current position.

Service Request

The Resume Service Request contains the following parameter:

1. CSTA Connection Identifier - indicates the Connection from which the message is to be resumed.

The Service Request also may include one or more of the following parameters:

2. Message Identifier - indicates the message to be resumed.
3. Duration - indicates the maximum duration of playback or recording.
4. CSTA Private Data Information.
5. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Resume Service causes the Voice Unit to resume playing or recording a previously suspended message at the current position. The current position in the message may have been changed by using the Reposition Service.

Voice.Review Method

The Review Service plays a portion of a voice message during a recording session. When this Service completes, the Voice Unit returns the message to the Suspend/Record state in its recording process. This does not affect the current position unless a Suspend occurs before the review completes.

Service Request

The Service Request contains the following parameters:

1. CSTA Connection Identifier - indicates the Connection on which the message is present.
2. Period - indicates the number of milliseconds to move backward and start reviewing. It also may have a value indicating "Beginning of Message".

The Service Request also may include one or more of the following parameters:

3. Message Identifier - indicates the message to review. If not provided, then the current message is used.
4. CSTA Private Data Information.
5. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Review Service will cause the Voice Unit to play the specified number of milliseconds before the current position in the currently-suspended, recorded message. When playback stops, the message will be at the same point it was when recording stopped. This allows recording to continue from the point at which it was interrupted.

Voice.SetVoiceAttribute Method

The Set Voice Attribute Service sets a voice attribute for a specified Connection and message.

Service Request

The Set Voice Attribute Service Request contains the following parameters:

1. CSTA Connection Identifier - indicates the Connection on which the voice attribute should be set.
2. Attribute - indicates the attribute to be set. This parameter has one of the following values:

   Speed - sets the playing speed. If requested, this reported value represents percentage of normal speed, with 100% indicating normal speed and the slowest reportable speed being 1%.

   Volume - sets the playing volume. The value is an integer from 0 to 100, with 100 indicating maximum volume and 0 indicating minimum volume.

   Level - sets the recording level. The value is an integer from 0 to 100, with 100 indicating maximum level and 0 indicating minimum level.

The Service Request includes the following parameter if there is more than one active message on the specified Connection. If there is only one active message on this Connection, then the following parameter may be sent.

3. Message Identifier - indicates the message whose attribute is being set. If this parameter is not provided, then the attribute of the currently-active message on the specified Connection is set.

The Service Request also may include one or more of the following parameters:

4. CSTA Private Data Information.
5. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Set Voice Attribute Service sets a voice attribute of a specified message.

Voice.Stop Method

The Stop Service stops the playing or recording of a message and resets the position pointer to the beginning of the message.

Service Request

The Stop Service Request contains the following parameter:

1. CSTA Connection Identifier - indicates the Connection to be handled.

The Service Request also may include one or more of the following parameters:

2. Message Identifier - indicates the message to be stopped.
3. CSTA Private Data Information.
4. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Stop Service causes the Voice Unit to stop playing or recording a message and will set the position pointer to the beginning of the message.

Voice.Suspend Method

The Suspend Service temporarily stops the playing or recording of the current message and sets a pointer to the current position in the message for later use.

Service Request

The Suspend Service Request contains the following parameter:

1. CSTA Connection Identifier - indicates the Connection on which to suspend playing or recording a message.

The Service Request also may include one or more of the following parameters:

2. Message Identifier - indicates the message to be suspended.
3. CSTA Private Data Information.
4. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement may include one or more of the following parameters:

1. CSTA Private Data Information.
2. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

The Suspend Service causes the Voice Unit to suspend playing or recording a message. While the message is suspended, the current position in the message can be changed by using the Reposition Service. The Voice Unit will continue to play or record the message at its current position in the message upon receiving the Resume Service Request.

Voice.SynthesizeMessage Method

The Synthesize Message Service constructs a voice message from a text message and returns the Message Identifier of the constructed message.

Service Request

The Synthesize Message Service Request contains the following parameter:

1. Text - provides the text to be spoken.

The Service Request also may include one or more of the following parameters:

2. Control Information - provides the parameters to control the speech synthesis.
3. CSTA Private Data Information.
4. CSTA Security Service Information.

Service Response

The server provides an acknowledgement of the Service Request. The acknowledgement is either positive or negative.

Positive acknowledgement

The positive acknowledgement includes the following parameter:

1. Message Identifier - indicates the message that was synthesized.

The positive acknowledgement also may include one or more of the following parameters:

2. CSTA Private Data Information.
3. CSTA Security Service Information.

Negative acknowledgement

The negative acknowledgement includes the following parameter:

1. CSTA Error Value - is one of the error values provided in CSTA Diagnostic Error.

Functional description

This is a “text to speech” function. If the Service Request is accepted, the Voice Unit constructs a voice message respecting the control parameters and returns its Message Identifier.

Event Reports

Event Report messages are sent from server to client when a monitor request has been positively acknowledged and a CSTA-reportable event has occurred. For monitors on the Switching Function, Event Reports are sent from the Switching Function to the Computing Function.

Service Response

Event Reports does not have responses.

Service Request

Event Reports reflect changes in object state(s) and may be accessed via monitors on devices or calls. The subject of the Event Report is specified in the Service Request.

Each Event Report contains a Cross Reference Identifier parameter, that uniquely identifies the monitor request that resulted in the Event Report. This parameter allows differentiating Event Reports resulting from multiple monitors. Event Reports include parameters that can have unknown or not required values. “Unknown” means that the parameter’s value is not known by the server. “Not required” means that the parameter’s value is that of the monitored device, when device-type monitoring is provided. In these cases, the parameter’s value can be unambiguously determined from the identity of the monitored device. The Last Redirecting Device parameter is a special case. If this parameter has the value “Not Required” then there has been no redirection.

Every Event Report may include the CSTA Private Data Information that can be used to convey information not defined in CSTA Standard. CSTA Private Data Information that changes the operation of the Service Request is not be provided in Event Reports. The CSTA Private Data parameter in Event Reports does never contain information that can be provided using other parameters, fields, and/or identifiers.

It is possible to augment the standard set of Event Reports to allow applications to identify a message as an Event Report even if the specific Event Report is not understood. Event Reports are generically identifiable as Event Reports.

Five categories of Switching Function Event Reports and an additional set of Event Reports that apply to Voice Units are defined in CSTA. These are: Agent State Event Report, Call Event Report, Feature Event Report, Maintenance Event Report, Voice Unit Event Report and Private Event Report. The category to which an Event Report belongs is always identifiable from the Event Report itself.

An instance of an Event Report Service Request may be referred to as an “Event Report”.

Functional Description

Event Report Service pictorials illustrate conditions before and after the change reported by the Event Report. All these figures illustrate only examples and are intended for clarification, and not to constrain implementations. The intent and use of these pictorials may be better understood with reference to the following example and keys:

Figure 28 – Example illustration for Service and Event Report descriptions

The figures illustrate conditions Before and After a successful Service or Event Report. In all such figures:

- boxes represent devices
- circles represent calls
- lines represent CSTA Connections between a call and a device
- absence of a line is equivalent to a CSTA Connection in the Null state
- labels in boxes and circles denote device and call instances, respectively
- labels on lines denote Connection states according to the following key:

  a = Alerting

c = Connected
f = Failed
h = Hold
i = Initiated
q = Queued
a/h = Alerting or Hold (other combinations work similarly)
***** = undefined

- grey boxes denote devices or calls that are unaffected by the Service/Event Report
- white boxes and circles denote devices and calls that are affected by the Service/Event Report

Call Event Reports

Call Event Reports are messages that indicate a change in state of one or more Connections in the switching sub-domain. One exception to the rule that Call Event Reports reflect state changes arises with regard to queuing. It is possible for a connection to queue successively without leaving the Queued state, generating a Call Event Report each time it is enqueued. Each Call Event Report may contain a parameter that summarizes the local Connection state from the perspective of a device that is a monitored object.

Every Call Event Report may contain one cause code to clarify the basic meaning of the Event Report. Call Event Report cause codes are defined in EventCauseType Enumeration.

Call Event Reports may apply to a single Connection, multiple Connections within a single call, or multiple Connections within multiple calls.

The following sub-clauses define the Call Event Reports and the information they provide. For each Call Event Report definition, the following two considerations apply:

- A Call Event Report indicates that the resultant state has been achieved independent of any previous state.
- Initial states are given purely for example to put the Call Event Report into a common telecommunications context.

Device.OnConferenced Event

Hierarchy

CTNGLib → Device → **OnConferenced**

Description

The Conferenced event indicates that two calls have been merget into one with no parties removed from the resulting call in process.

The conferenced report indicates a three-party conference has been established.

Syntax

```
device.OnConferenced primaryOldCall, secondaryOldCall, confController,
      addedParty, conferenceConnections, localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

primaryOldCall

String, by value, input, indicates the Device ID and Call ID for the cleared connection.

secondaryOldCall

String, by value, input, indicates the Device ID of the clearing device.

confController

String, by value, input, indicates a Device ID and Call ID for the previous call that this device was involved in. For the conference leader, this represents the previously held call.

addedParty

String, by value, input, indicates the Device ID and Call ID of the secondary call for this device. For the conference leader, this represents the connected call. For conference members, this field is empty.

conferenceConnections

String, by value, input, indicates a list of Call and Device IDs for all parties in the conference.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, input, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, No cause codes are provided.

Functional Description

This Event Report indicates that two calls have been merged into one with no parties removed from the resulting call in the process.

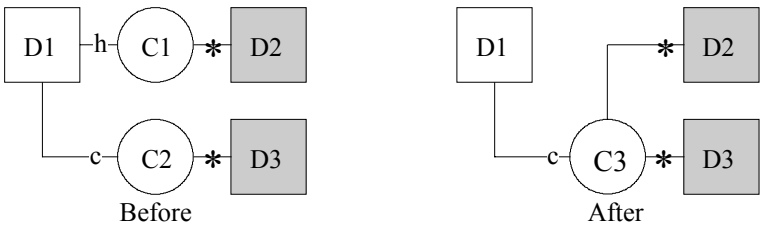

Figure 29 – Conferenced

This Conferenced Event Report applies to multiple Connections within multiple calls.

This Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates a primary known call that was conferenced.
2. CSTA Device Identifier - indicates the device that conferenced the call. If a device is not specified, then this parameter indicates “Unknown” or “Not Required”.
3. CSTA Device Identifier - indicates the device last added to the call. If a device is not specified, then this parameter indicates “Unknown” or “Not Required”. The last added device is the last device to have been called or answered by a device forming or enlarging a conference. The use of Hold or Retrieve Services during call set-up does not alter the designation of the last added device.

The Event Report includes the following parameter if previous Event Reports have provided this pre-conference Connection Identifier:

4. CSTA Connection Identifier - indicates a secondary call, if known, that was conferenced.

The Event Report also may include one or more of the following parameters:

list the following three parameters for one or more endpoints in the resultant call:

5. CSTA Connection Identifier - indicates the Connection Identifier for the endpoint in the resultant call.
6. CSTA Device Identifier - optional. If provided, it is the Device Identifier for the endpoint.
7. CSTA Connection Identifier - indicates the previous Connection Identifier for the party in its original call.
8. Local Connection State - if this Event Report was generated from a device monitor then this parameter indicates the Connection state for this call at the monitored device.
9. CSTA Application Correlator Data - provides application-specific data associated with the call.
10. Cause - indicates a reason for the Event Report.
11. CSTA Private Data Information.
12. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Active Monitor, Single Step Conference, Key Operation, New Call, Override, Recall, and Silent Monitor.

Remarks

Note: A new Call ID is created when an conference is established.

See Also

Applies To: Device Class

Device.OnCallCleared Event

Hierarchy

CTNGLib → Device → **OnCallCleared**

Description

This event indicates that a call has been cleared. Normally this occurs when the last remaining device disconnects from the call. It can also occur when a call is immediately dissolved - for example when a conference call is dissolved by the conference controller.

Syntax

```
device.OnCallCleared clearedCall, localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

clearedCall

String, by value, input, indicates the Call ID for the cleared connection.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, input, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, No cause codes are provided.

The Event Report includes the following parameter:

1. CSTA Connection Identifier - indicates the call that was cleared.

The Event Report also may include one or more of the following parameters:

2. Local Connection State - if this Event Report was generated from a device monitor then this parameter indicates that the Connection state for this call at the monitored device is Null.
3. CSTA Application Correlator Data - provides application-specific data associated with the call.
4. Cause - indicates a reason for the Event Report.
5. CSTA Private Data Information.
6. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Call Back, Call Cancelled, Call Not Answered, Incompatible Destination, Invalid Account Code, Key Operation, Maintenance, Overflow, Override, and Resources not Available.

Functional Description

This Event Report indicates that a call has been cleared. Normally this occurs when the last remaining device disconnects from the call. It can also occur when a call is immediately dissolved - for example when a conference call is dissolved by the conference controller.

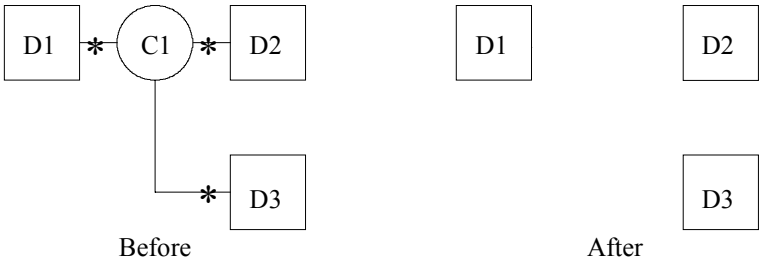

Figure 30 – Call Cleared

The Call Cleared Event Report applies to all Connections within a call.

Remarks

MD110 Specific: Not supported by MD110.

See Also

Applies To: Device Class

Device.OnConnectionCleared Event

Hierarchy

CTNGLib → Device → **OnConnectionCleared**

Description

The Connection Cleared event indicates one device has cleared from the call. For a multiple party call, the call may still exist.

Syntax

```
device.OnConnectionCleared droppedConnection, releasingDevice,
      localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

droppedConnection

String, by value, input, indicates the Device ID and Call ID for the cleared connection.

releasingDevice

String, by value, input, indicates the Device ID of the clearing device.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, input, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, No cause codes are provided.

Functional Description

This Event Report indicates that a device in a call disconnects or is dropped from the call. This Event Report, however, does not indicate that a transferring device has left a call in the act of transferring that call.

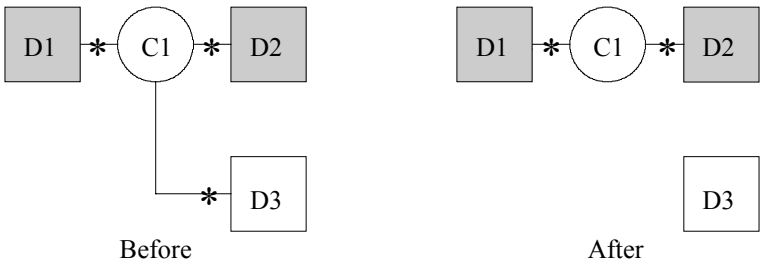

Figure 31 – Connection Cleared

The Connection Cleared Event Report applies to a single Connection.

This Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the Connection that was dropped from the call.
2. CSTA Device Identifier - indicates the device that dropped from the call. If a device is not specified then the parameter indicates "Unknown" or "Not Required".

The Event Report also may include one or more of the following parameters:

3. Local Connection State - if this Event Report was generated from a device monitor then this parameter indicates the Connection state for this call at the monitored device.

4. CSTA Application Correlator Data - provides application-specific data associated with the call.
5. Cause - indicates a reason for the Event Report.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Call Back, Call Cancelled, Call Not Answered, Destination Not Obtainable, Do Not Disturb, Incompatible Destination, Key Operation, Overflow, Override, Park, and Resources Not Available.

Remarks

None.

Examples

Visual Basic Sample

```
Option Explicit
Public csta As New CTNGLib.System
Public pabx As New CTNGLib.pabx
Public WithEvents dev As CTNGLib.device

Private Sub Initialize
    csta.AddPABX "md110", "csta.md110.com", 2555
    pabx.Open "md110"
    Set dev = New CTNGLib.Device
    dev.Open pabx, "805"
End Sub

Private Sub dev_OnDelivered( _
    ByVal connection As String, _
    ByVal alertingDevice As String, _
    ByVal callingDevice As String, _
    ByVal calledDevice As String, _
    ByVal lastRedirectionDevice As String, _
    ByVal localConnectionInfo As CTNGLib.LocalConnectionStateType, _
    ByVal cause As CTNGLib.EventCauseType _
)
    MsgBox "Call received from " + callingDevice
End Sub
```

See Also

Applies To: Device Class

Device.OnDelivered Event

Hierarchy

CTNGLib → Device → **OnDelivered**

Description

The Delivered event indicates a device is ringing another device or has received an incoming call.

Syntax

```
device.OnDelivered connection, alertingDevice, callingDevice,
      calledDevice, lastRedirectionDevice, localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

connection

String, by value, input, indicates a Device ID and Call ID for the ringing connection.

alertingDevice

String, by value, input, indicates the Device ID of the ringing device. For an outgoing call, thiw will be the actual B-party digits, if known.

callingDevice

String, by value, input, indicates a Device ID of the calling device. For an incoming call, thiw will be the actual A-party digits, if known.

calledDevice

String, by value, input, indicates the Device ID of the originally called device. If, for example, a diversion occurred or the call is an ACD call, this value would be the ACD group number.

lastRedirectionDevice

String, by value, input, no value is provided.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, input, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, the cause of the event. The following cause codes could happen (see Table 22 - EventCauseType Enumeration Values):

- ec_campOn
- ec_newCall

Functional Description

This Event Report indicates that "alerting" (tone, ring, etc.) is being applied to a device or that the server has detected that alerting has been applied to a device.

Figure 32 – Delivered

This Event Report applies to a single Connection.

CSTA servers may allow multiple devices to be alerted at the same time. When this happens, a follow-on Call Established Event Report for the call might have a different Connected CSTA Device Identifier than the Called CSTA Device Identifier passed in the previous Delivered Event Report for that call. In addition, every time a Delivered Event Report for a call is followed by another Delivered Event Report for that call at another device, it may imply that the first device is no longer involved in the call and that the call may have been redirected.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the Connection that is Alerting.
2. CSTA Device Identifier - indicates the device that is Alerting. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
3. CSTA Device Identifier - indicates the calling device. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
4. CSTA Device Identifier - indicates the originally called device. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
5. CSTA Device Identifier - indicates the previously alerted device. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.

The Event Report also may include one or more of the following parameters:

6. CSTA Connection Identifier - indicates the Connection of the originating device. For an incoming, external call, the originating device (e.g. trunk) identifies the device whereby the call entered the switching sub-domain. For an outgoing, external call, the originating device represents the device whereby a call exits the switching sub-domain.
7. Local Connection State - if the monitor request was for a device, this parameter indicates the Connection state of the monitored device for this call.
8. CSTA Application Correlator Data - provides application-specific data associated with the call.
9. Cause - indicates a reason for the Event Report.
10. CSTA Private Data Information.
11. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Call Back, Call Forward, Call Fd.-Immediate, Call Fd.-Busy, Call Fd.-No Answer, Camp On, Key Operation, New Call, No Available Agents, Overflow, Override, Recall, Redirected, Entering Distribution, Distributed, Transfer, and Single Step Transfer.

Remarks

MD110 Specific: Customer ID is added as Private Data in this event, if associated to the call.

Examples

Visual Basic Sample

```
Option Explicit
Public csta As New CTNGLib.System
Public pabx As New CTNGLib.pabx
Public WithEvents dev As CTNGLib.device

Private Sub Initialize
    csta.AddPABX "md110", "csta.md110.com", 2555
    pabx.Open "md110"
    Set dev = New CTNGLib.Device
    dev.Open pabx, "805"
End Sub

Private Sub dev_OnDelivered( _
    ByVal connection As String, _
    ByVal alertingDevice As String, _
```

```
        ByVal callingDevice As String, _
        ByVal calledDevice As String, _
        ByVal lastRedirectionDevice As String, _
        ByVal localConnectionInfo As CTNGLib.LocalConnectionStateType, _
        ByVal cause As CTNGLib.EventCauseType _
    )
        MsgBox "Call received from " + callingDevice
    End Sub
```

See Also

Applies To: Device Class

Device.OnDiverted Event

Hierarchy

CTNGLib → Device → **OnDiverted**

Description

The Diverted event indicates that a call has been diverted from a monitored device. The event report is provided for the device from which the call is diverted.

Syntax

```
device.OnDiverted connection, divertingDevice, newDestination,
      localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

connection

String, by value, input, indicates the Device ID and Call ID for the connection that the monitored device was involved in.

divertingDevice

String, by value, input, indicates the Device ID of the device from which the call was diverted.

newDestination

String, by value, input, indicates a Device ID of the device to which the call is to be diverted, if known.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, input, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, the cause of the event. The following cause codes could happen (see Table 22 - EventCauseType Enumeration Values):

- ec_campOn
- ec_newCall

Functional Description

This Event Report indicates that a call has been diverted from a monitored device and that the call is no longer present at the device. This Event Report is provided only from monitors active on the device from which the call is diverted.

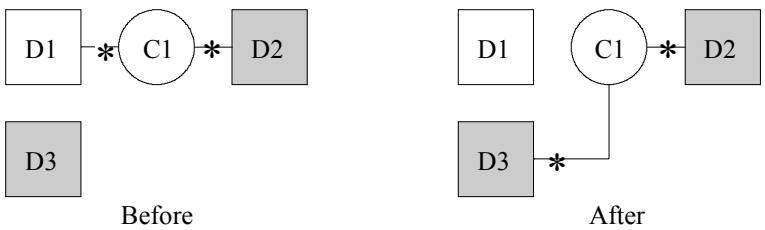

Figure 33 – Diverted

This Event Report applies to a single Connection.

If the call alerted the device, then the Event Report includes the following parameter. If the call did not alert the device, then the Event Report may include the following parameter:

1. CSTA Connection Identifier - indicates the Connection that was alerting or that was diverted from the monitored device.

The Event Report includes the following parameters:

2. CSTA Device Identifier - indicates the device from which the call was Diverted. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
3. CSTA Device Identifier - indicates the device to which the call was diverted. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.

The Event Report also may include one or more of the following parameters:

4. Local Connection State - if the Event Report was from a device monitor, this parameter indicates the Connection state of this call at the monitored device.
5. CSTA Application Correlator Data - provides application-specific data associated with the call.
6. Cause - indicates a reason for the Event Report.
7. CSTA Private Data Information.
8. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Call Forward, Call Fd.-Immediate, Call Fd.-Busy, Call Fd.-No Answer, Call Not Answered, Call Pickup, Do Not Disturb, Incompatible Destination, Key Operation, No Available Agents, Overflow, Recall, Redirected, and Voice Unit Initiator.

Remarks

Note: A new Call ID is created when an conference is established.

See Also

Applies To: Device Class

Device.OnEstablished Event

Hierarchy

CTNGLib → Device → **OnEstablished**

Description

The Established event indicates one of the following:

- a call has been answered
- a call has been picked up
- a 3-party conference is now a 2-party call

Syntax

```
device.OnEstablished establishedConnection, answeringDevice,
      callingDevice, lastRedirectionDevice, localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

establishedConnection

String, by value, input, indicates the Device ID and Call ID for the connection joining the call.

answeringDevice

String, by value, input, indicates the Device ID of the joining device. For an ougoing call, thiw will be the actual B-party digits, if known.

callingDevice

String, by value, input, indicates a Device ID of the calling device. For an incoming call, this will be the actual A-party digits, if known.

lastRedirectionDevice

String, by value, input, no value is provided.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, input, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, no value is provided.

Functional Description

This Event Report indicates that the server has detected that a device has answered or connected to a call.

Figure 34 – Established

This Event Report applies to a single Connection.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the Connection that joined the call.
2. CSTA Device Identifier - indicates the device that joined the call. If a device is not specified then this parameter indicates "Unknown" or "Not Required".

3. CSTA Device Identifier - indicates the calling device. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
4. CSTA Device Identifier - indicates the originally called device. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
5. CSTA Device Identifier - indicates the previously alerted device. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.

The Event Report also may include one or more of the following parameters:

6. CSTA Connection Identifier - indicates the Connection of the originating device. For an incoming, external call, the originating device (e.g. trunk) identifies the device whereby the call entered the switching sub-domain. For an outgoing, external call, the originating device represents the device whereby a call exits the switching domain.
7. Local Connection State - if the Event Report was generated from a device monitor, this parameter indicates the Connection state of the call at the monitored device.
8. CSTA Application Correlator Data - provides application-specific data associated with the call.
9. Cause - indicates a reason for the Event Report.
10. CSTA Private Data Information.
11. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Alternate, Call Pickup, Single Step Conference, Key Operation, New Call, Override, Recall, and Transfer.

Remarks

MD110 Specific: Customer ID is added as Private Data in this event, if associated to the call.

See Also

Applies To: Device Class

Device.OnFailed Event

Hierarchy

CTNGLib → Device → **OnFailed**

Description

The Failed event indicates that a call dialed from a device has failed due to a number of reasons such as invalid digits dialed, the dialed party is busy etc.

Syntax

```
device.OnFailed failedConnection, failedDevice, calledDevice,
      localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

failedConnection

String, by value, input, indicates the Device ID and Call ID for the failing connection. If the failing device is unknown, only the Call ID will be provided.

failedDevice

String, by value, input, indicates the Device ID of the failing device, if known.

calledDevice

String, by value, input, indicates a Device ID of the called device, if known.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, input, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, the cause of the event. The following cause values could happen (see Table 22 - EventCauseType Enumeration Values):

- ec_destNotObtainable
- ec_doNotDisturb
- ec_lockout
- ec_reorderTone
- ec_callBack
- ec_noAvailableAgents
- ec_networkCongestion
- ec_incompatibleDestination
- ec_resourceNotAvailable

Functional Description

This Event Report indicates that a call cannot be completed or when a Connection enters the Failed state for other reasons.

Figure 35 – Failed

The Failed Event Report applies to a single Connection.

The Failed Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the Connection that failed.
2. CSTA Device Identifier - indicates the device that failed. If a device is not specified then the parameter indicates “Unknown” or “Not Required”.
3. CSTA Device Identifier - indicates the called device. If a device is not specified then the parameter indicates “Unknown” or “Not Required”.

The Event Report also may include one or more of the following parameters:

4. Local Connection State - if the Event Report was generated from a device then this parameter indicates the Connection state of this call at the monitored device.
5. CSTA Application Correlator Data - provides application-specific data associated with the call.
6. Cause - indicates a reason for the Event Report.
7. CSTA Private Data Information.
8. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Alternate, Busy, Call Cancelled, Call Forward, Call Fd.-Immediate, Call Fd.-Busy, Call Fd.-No Answer, Call Not Answered, Camp On, Destination Not Obtainable, Do Not Disturb, Incompatible Destination, Invalid Account Code, Key Operation, Lockout, Maintenance, Network Congestion, Network Not Obtainable, No Available Agents, Overflow, Override, Recall, Redirected, Reorder Tone, Resources Not Available, Transfer, and Trunks Busy.

Remarks

MD110 Specific: Customer ID is added as Private Data in this event, if associated to the call.

See Also

Applies To: Device Class

Device.OnHeld Event

Hierarchy

CTNGLib → Device → **OnHeld**

Description

The Held event indicates that device has been held or is holding another device.

This event applies to a single connection.

Syntax

```
device.OnHeld heldConnection, holdingDevice, localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

heldConnection

String, by value, input, indicates the Device ID and Call ID where the hold was activated.

holdgingDevice

String, by value, input, indicates the Device ID of the device where the hold was activated.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, input, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, no value is provided.

Functional Description

This Event Report indicates that the server detected that communication on an existing call has been temporarily interrupted at one of the devices on the call.

Figure 36 – Held

The Held Event Report applies to a single Connection.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the Connection at which hold was activated.
2. CSTA Device Identifier - indicates the device at which hold was activated. If a device is not specified then the parameter indicates "Unknown" or "Not Required".

The Event Report also may include one or more of the following parameters:

3. Local Connection State - if the Event Report was generated from a device monitor then this parameter indicates the Connection state of this call at the monitored device.
4. CSTA Application Correlator Data - provides application-specific data associated with the call.
5. Cause - indicates a reason for the Event Report.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Alternate, Call Forward, Call Fd.-No Answer, Key Operation, Recall, and Transfer.

Remarks

None.

See Also

Applies To: Device Class

Device.OnNetworkReached Event

Hierarchy

CTNGLib → Device → **OnNetworkReached**

Description

The Network Reached event indicates that a trunk device has been seized for an outgoing call.

Syntax

```
device.OnNetworkReached connection, trunkUsed, calledDevice,
      localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

connection

String, by value, input, indicates the Device ID and Call ID for the outbound connection.

trunkUsed

String, by value, input, no value is provided.

calledDevice

String, by value, input, no value is provided.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, input, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, no value is provided.

Functional Description

This Event Report indicates that a call has cut through the CSTA switching sub-domain boundary to another network (e.g. has reached an outgoing trunk). This Event Report implies that there may be a reduced level of Event Reporting and possibly no additional device feedback, except disconnect/drop, provided for this party in the call. Additionally, the CSTA application should assume that it cannot directly manipulate the far-end device represented by the trunk. A Network Reached Event Report is never sent for calls made to devices connected directly to the CSTA switching sub-domain. This event indicates that a Connection with a trunk has reached the Connected state, and that further events for that Connection refer to the state of the endpoint to which the trunk is a gateway.

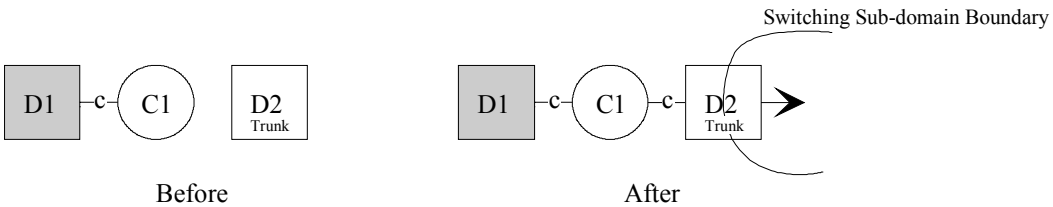

Figure 37 – Network Reached

This Network Reached Event Report applies to a single Connection.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the outbound Connection (of the trunk) to another network.

2. CSTA Device Identifier - indicates the trunk that was selected. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
3. CSTA Device Identifier - indicates the destination device. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.

The Event Report also may include one or more of the following parameters:

4. Local Connection State - if the Event Report was generated from a device monitor then this parameter indicates the Connection state for the call at the monitored device.
5. CSTA Application Correlator Data - provides application-specific data associated with the call.
6. Cause - indicates a reason for the Event Report.
7. CSTA Private Data Information.
8. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Call Forward, Call Fd.-Immediate, Call Fd.-Busy, Call Fd.-No Answer, Key Operation, Overflow, Redirected, Resources Not Available, and Transfer.

Remarks

None.

See Also

Applies To: Device Class

Device.OnOriginated Event

Hierarchy

CTNGLib → Device → **OnOriginated**

Description

The Originated event indicates that the server is attempting to make a call; it implies that input activity is complete and that a call (rather than feature) has been requested. This event also provides the Call ID assigned to the call.

Syntax

```
device.OnOriginated originatedConnection, callingDevice, calledDevice,
      originatingDevice, localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

originatedConnection

String, by value, input, indicates the Device ID and Call ID for the dialing device.

callingDevice

String, by value, input, indicates the device originating the call.

calledDevice

String, by value, input, no value is provided.

originatingDevice

String, by value, input, no value is provided.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, no value is provided.

Functional Description

This Event Report indicates that the server is attempting to make a call. It implies that input activity is complete and that a call (rather than feature) has been requested.

Figure 38 – Originated

This Originate Event Report applies to a single Connection.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the Connection at which the call originated.
2. CSTA Device Identifier - indicates the calling device. For an incoming call, the calling device identifies the party outside the switching sub-domain that placed the call. For an outgoing call, the calling device represents the party on behalf of whom the call is originated. If a device is not specified then this parameter indicates "Unknown" or "Not Required".

3. CSTA Device Identifier - indicates the called device. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.

The Event Report also may include one or more of the following parameters:

4. CSTA Device Identifier - indicates the originating device. For an incoming call, the originating device (e.g. trunk) identifies the device whereby the call entered the switching sub-domain. For an outgoing call, the originating device represents the device whereby a call exits the switching domain. This parameter is only given if the call crosses the switching sub-domain boundary.
5. Local Connection State - if the Event Report was generated from a device monitor then this parameter indicates that the Connection state for this call at the monitored device is Connect.
6. CSTA Application Correlator Data - provides application-specific data associated with the call.
7. Cause - indicates a reason for the Event Report.
8. CSTA Private Data Information.
9. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Call Back, Key Operation, New Call, Override, Make Call, and Silent Monitor.

Remarks

None.

See Also

Applies To: Device Class

Device.OnQueued Event

Hierarchy

CTNGLib → Device → **OnQueued**

Description

The Queued event indicates that a monitored device has been queued or a call has been queued at a monitored ACD queue.

Syntax

```
device.OnQueued queuedConnection, queue, callingDevice, calledDevice,
      lastRedirectionDevice, localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

queuedConnection

String, by value, input, indicates the Device ID and Call ID for the queued connection.

queue

String, by value, input, indicates the Device ID of the device that the call has been queued to.

callingDevice

String, by value, input, indicates the Device ID of the calling device, if known.

calledDevice

String, by value, input, indicates the Device ID of the called device. This is the actual B-party digits, if known. If they are not known, this field represents the queue Device ID.

lastRedirectionDevice

String, by value, input, no value is provided.

numberQueued

String, by value, input, no value is provided.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, the cause of the event. The following cause values could happen (see Table 22 - EventCauseType Enumeration Values):

- ec_callBack
- ec_newCall

Functional Description

This Event Report indicates that a call queued. Queuing may occur, for example, at an ACD, hunt group, or other device. Queuing also may occur during network routeing without an associated device.

Figure 39 – Queued

The Queued Event Report applies to a single Connection.

This Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the Connection that queued.
2. CSTA Device Identifier - indicates the queuing device (if queued at a device). If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
3. CSTA Device Identifier - indicates the calling device. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
4. CSTA Device Identifier - indicates the called device. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
5. CSTA Device Identifier - indicates the redirecting device if the call was redirected. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.

The Event Report also may include one or more of the following parameters:

6. Number - indicates the total number of calls in queue, including this call.
7. Number - indicates the number of calls ahead of the call when it was enqueued (the enqueued call is not counted in this number).
8. Local Connection State - if the Event Report was generated from a device monitor then this parameter indicates the Connection state of this call at the monitored device.
9. CSTA Application Correlator Data - provides application-specific data associated with the call.
10. Cause - indicates a reason for the Event Report.
11. CSTA Private Data Information.
12. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Busy, Call Back, Call Forward, Call Fd.-Immediate, Call Fd.-Busy, Call Fd.-No Answer, Camp On, Destination Not Obtainable, Do Not Disturb, Key Operation, Network Congestion, Network Not Obtainable, No Available Agents, Overflow, Park, Recall, Redirected, Resources Not Available, Transfer, and Trunks Busy.

Remarks

MD110 Specific: Customer ID is added as Private Data in this event, if associated to the call.

See Also

Applies To: Device Class

Device.OnRetrieved Event

Hierarchy

CTNGLib → Device → **OnRetrieved**

Description

The Retrieved event indicates that a previously held call has been retrieved and is now connected.

Syntax

```
device.OnRetreived retrievedConnection, retrievingDevice,
      localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

retrievedConnection

String, by value, input, indicates the Device ID and Call ID for the retrieved connection.

retrievingDevice

String, by value, input, indicates the Device ID of the retrieving device.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, the cause of event.

Functional Description

This Event Report indicates that the server detected that a previously held call has been retrieved.

Figure 40 – Retrieved

The Retrieved Event Report applies to a single Connection.

This Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the Connection on which hold was de-activated.
2. CSTA Device Identifier - indicates the device at which hold was de-activated. If a device is not specified then this parameter indicates "Unknown" or "Not Required".

The Event Report also may include one or more of the following parameters:

3. Local Connection State - if the Event Report was generated from a device monitor then this parameter indicates the Connection state of this call at the monitored device.
4. CSTA Application Correlator Data - provides application-specific data associated with the call.
5. Cause - indicates a reason for the Event Report.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Alternate, Key Operation, Recall, and Transfer.

Remarks

Note.

See Also

Applies To: Device Class

Device.OnServiceInitiated Event

Hierarchy

CTNGLib → Device → **OnServiceInitiated**

Description

The Service Initiated indicates that a device has gone offhook manually (i.e., not by a Make Call request). This event also provides the Call ID assigned to the call. This report is typically issued when a "dial-tone" is detected. The Service Initiated event indicates that:

- a call may be originated, or
- a feature may be invoked.

The Service Initiated event report applies to a single connection.

Syntax

```
device.OnServiceInitiated initiatedConnection, localConnectionInfo,
       cause
```

Parameters

This event will include the following parameters:

initiatedConnection

String, by value, input, indicates the Device ID and Call ID for the device going offhook.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, the cause of event.

Functional Description

This Event Report indicates that telecommunications service has been initiated at a monitored device. The server typically issues this Event Report when "dial tone" is provided. This Event Report indicates that either a call may be originated or a feature may be invoked. This event also may indicate that a device is prompting a user to start a call.

Note 6:

The Make Call Service will often prompt a user to take a phone off-hook if that phone is not able to do so without manual intervention. In this case the Service Initiated event may be used to indicate that the phone is prompting the user to go off-hook. In this case the Make Call or Call Back cause code are used.

Note 7:

This Event Report may not be sent for functional (en-bloc BRI) terminals and may not be sent for calls that are set up without receiving dial tone or other prompting, like CSTA calls initiated with Make Call Service from a hands-free telephone.

Figure 41 – Service Initiated

The Service Initiated Event Report applies to a single Connection.

The Event Report includes the following parameter:

1. CSTA Connection Identifier - indicates the Connection at which service has been established. This Connection Identifier continues to be used if the initiated service becomes a Call.

The Event Report also may include one or more of the following parameters:

2. Local Connection State - if the Event Report was generated from a device monitor then this parameter indicates the Connection state for this call at the monitored device Initiated.
3. Cause - indicates a reason for the Event Report.
4. CSTA Private Data Information.
5. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Call Back, Call Cancelled, Key Operation, Make Call, Recall, and Override.

Remarks

Note.

See Also

Applies To: Device Class

Device.OnTransferred Event

Hierarchy

CTNGLib → Device → **OnTransferred**

Description

The Transferred event indicates that a device has transferred the held party to the ringing or connected party.

Syntax

```
device.OnTransferred primaryOldCall, secondaryOldCall,
      transferringDevice, transferredDevice, transferredConnections,
      localConnectionInfo, cause
```

Parameters

This event will include the following parameters:

primaryOldCall

String, by value, input, indicates the Device ID and Call ID for the previous call that this device was involved in.

secondaryOldCall

String, by value, input, indicates the Device ID and Call ID of the secondary call for this device.

For the transferring device, this call is the previously active call.

For devices other than the transferring device, this field is not provided.

transferringDevice

String, by value, input, indicates the Device ID of the device that transferred the call.

transferredDevice

String, by value, input, indicates the Device ID of the device to which the call was transferred.

transferredConnections

String, by value, input, indicates a list of Call and Device IDs for the devices remaining in the call after the transfer.

localConnectionInfo

CTNGLib.LocalConnectionStateType, by value, indicates the state of this device's connection.

cause

CTNGLib.EventCauseType, by value, input, the cause of event.

Functional Description

This Event Report indicates that an existing call was transferred to another device and that the device requesting the transfer has been dropped from the call. The transferring device does not appear in any future Event Reports for the call.

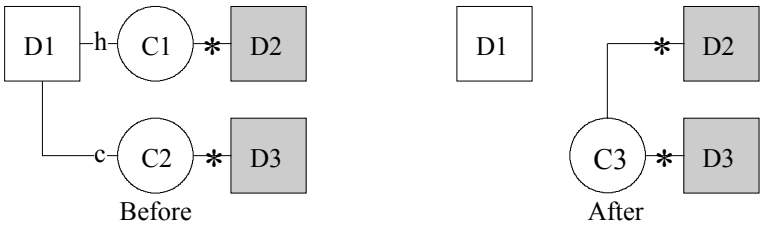

Figure 42 – Transferred

The Transferred Event Report applies to single Connections in multiple calls.

This Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates a primary known call that was transferred.
2. CSTA Device Identifier - indicates the device that transferred the call (D1 in the preceding figure). If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
3. CSTA Device Identifier - indicates the device (D3 in the preceding figure) to which the call (C1 in the preceding figure) was transferred. If a device is not specified then the parameter indicates “Unknown” or “Not Required”.

This Event Report includes the following parameter if previous Event Reports have provided this pre-transfer Connection Identifier:

4. CSTA Connection Identifier - indicates a secondary call, if known, that was transferred.

The Event Report also may include any one or more of the following parameters:

list the following three parameters for every endpoint in the resultant call:

5. CSTA Connection Identifier - indicates the Connection Identifier for the endpoint in the resultant call.
6. CSTA Device Identifier - if provided, this parameter indicates the Device Identifier for the endpoint.
7. CSTA Connection Identifier - indicates the previous Connection Identifier for the endpoint in its original call.
8. Local Connection State - if the Event Report was generated from a device monitor then this parameter indicates the Connection state of this call at the monitored device.
9. CSTA Application Correlator Data - provides application-specific data associated with the call.
10. Cause - indicates a reason for the Event Report.
11. CSTA Private Data Information.
12. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Key Operation, New Call, Recall, Redirected, Transfer, and Voice Unit Initiator.

Remarks

Note: A transfer results in the generation of a new Call ID for the resultant call.

MD100 Specific: Customer ID is added as Private Data in this event, if associated to the call.

See Also

Applies To: Device Class

Agent State Event Reports

An Agent State Event Report is a message that indicates a change in state of an agent in the CSTA network.

Agent State Event Reports always indicates the new state an agent enters independent of any previous state.

An Agent State Event Report includes the Device Identifier of the agent device to which it applies.

Every Agent State Event Report also may include CSTA Private Data Information.

Transitions between Agent States give rise to Agent Events that can be reported in Agent Event Reports. The following table indicates how particular Agent Event Reports might arise from these transitions, based on the Agent State model given here. Where a cell contains two events, this means that both events will be generated by the indicated state transition. A Switching Function with a different underlying Agent State model might produce a different pattern of Agent Event Reports. In the following table, row headings represent initial Agent States and column headings represent the state into which the agent enters. Cell entries indicate the Agent Event Report that would arise. The Agent Event Reports are more fully defined in the subsequent Agent Event Report sub-clauses.

Table 2 - Agent State transitions and resulting Agent Events

	AGENT NULL	AGENT NOT READY	AGENT READY	AGENT BUSY	AGENT WORKING AFTER CALL
AGENT NULL		Agent_logged_on Agent_not_ready	Agent_logged_on Agent_ready		
AGENT NOT READY	Agent_logged_off		Agent_ready		
AGENT READY	Agent_logged_off	Agent_not_ready		Agent_busy	
AGENT BUSY		Agent_not_ready	Agent_ready		Agent_working_ after_call
AGENT WORKING AFTER CALL	Agent_logged_off	Agent_not_ready	Agent_ready		

Device.OnAgentLoggedOff Event

This Event Report indicates that an agent has logged off from a group or device. It implies that the agent cannot accept any additional ACD calls to the group or device.

This Event Report includes the following parameter:

1. CSTA Device Identifier - indicates the device from which the agent logged off. If a device is not specified then this parameter indicates “Not Required”.

The Event Report also may include one or more of the following parameters:

2. CSTA Agent Identifier - indicates the Agent Identifier.
3. CSTA Device Identifier - indicates the group or pilot from which the agent is logging out.
4. Password Data - indicates the agent password for logging off.
5. Cause - indicates a reason for the Event Report.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Device.OnAgentLoggedOn Event

This Agent State Event Report indicates that an agent has logged on and is ready to contribute to the activities of the group or device to which the agent has logged in. It does not indicate that the agent is ready to accept ACD calls.

This Event Report includes the following parameter:

1. CSTA Device Identifier - indicates the device from which the agent logged on. If a device is not specified then this parameter indicates “Not Required”.

The Event Report also may include one or more of the following parameters:

2. CSTA Agent Identifier - indicates the Agent Identifier.
3. CSTA Device Identifier - indicates the group or pilot to which the agent is logging in.
4. Password Data - indicates the agent password for logging in.
5. Cause - indicates a reason for the Event Report.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Device.OnAgentNotReady Event

This Event Report indicates that an agent is occupied with some task other than serving an ACD call. It implies that the agent cannot accept additional ACD calls but may receive non-ACD calls.

This Event Report includes the following parameter:

1. CSTA Device Identifier - indicates the device at which Not Ready was invoked. If a device is not specified then the parameter indicates “Not Required”.

The Event Report also may include one or more of the following parameters:

2. CSTA Agent Identifier - indicates the Agent Identifier.
3. Cause - indicates a reason for the Event Report.
4. CSTA Private Data Information.
5. CSTA Security Service Information.

Device.OnAgentReady Event

This Event Report indicates that an agent is ready to accept ACD calls even though they may be busy with a non-ACD call. Call Event Reports can provide information on involvement with both ACD and non-ACD calls.

The Event Report includes the following parameter:

1. CSTA Device Identifier - indicates the device at which Agent Ready was invoked. If a device is not specified then the parameter indicates “Not Required”.

The Event Report also may include one or more of the following parameters:

2. CSTA Agent Identifier - indicates the Agent Identifier.
3. Cause - indicates a reason for the Event Report.
4. CSTA Private Data Information.
5. CSTA Security Service Information.

Device.OnAgentWorkingAfterCall Event

This Event Report indicates that an agent is occupied with serving an ACD call. It implies that the agent is no longer connected to the call but is still occupied with work related to that call. It also implies that the agent is not able to accept additional ACD calls but may receive non-ACD calls.

The Event Report includes the following parameter:

1. CSTA Device Identifier - indicates the device at which Agent Working After Call was invoked. If a device is not specified then the parameter indicates “Not Required”.

The Event Report also may include one or more of the following parameters:

2. CSTA Agent Identifier - indicates the Agent Identifier.
3. Cause - indicates a reason for the Event Report.
4. Group - indicates the group for which the agent is performing after call work.
5. CSTA Private Data Information.
6. CSTA Security Service Information.

Device.OnAgentBusy Event

This Event Report indicates that an agent is occupied with serving an ACD call. It implies that the agent is connected to the ACD call and is occupied with work belonging to that ACD call. It also implies that the agent may be able to accept non-ACD calls.

The Event Report includes the following parameter:

1. CSTA Device Identifier - indicates the device at which Agent Busy was invoked. If a device is not specified then the parameter indicates “Not Required”.

The Event Report also may include one or more of the following parameters:

2. CSTA Agent Identifier - indicates the Agent Identifier.
3. Cause - indicates a reason for the Event Report.
4. Group - indicates the group in which the agent is busy.
5. CSTA Private Data Information.
6. CSTA Security Service Information.

Feature Event Reports

Each Feature Event Report is a message that indicates a change in Feature state of a call or device in the CSTA network. Like Call Event Reports, each Feature Event Report indicates the new state that the feature enters independent of any previous state.

Every Event Report includes the identifier of the call or device to which the event applies.

Device.OnCallInformation Event

This Event Report indicates that an account code feature has collected data for a party on a call.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the party that entered the account code and the call for which it was entered.
2. CSTA Device Identifier - indicates the device at which the account code was entered. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.

The Event Report also may include one or more of the following parameters:

3. Account Code Data - indicates the account code to associate with the call.
4. Authorization Code Data - indicates the authorization code to allow the call.
5. CSTA Application Correlator Data - provides application-specific data associated with the call.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Device.OnDoNotDisturb Event

This Event Report indicates that the Do Not Disturb feature has been invoked for a device.

The Do Not Disturb Event Report includes the following parameters:

1. CSTA Device Identifier - indicates the device at which Do Not Disturb was invoked. If a device is not specified then this parameter indicates “Not Required”.
2. Flag - indicates whether the feature was turned on or off.

This Event Report also may include one or more of the following parameters:

3. CSTA Private Data Information.
4. CSTA Security Service Information.

Device.OnForwarding Event

This Event Report indicates that the Forwarding feature has been invoked for a device.

The Forwarding Event Report includes the following parameters:

1. CSTA Device Identifier - indicates the device at which Forwarding was invoked. If a device is not specified then this parameter indicates “Not Required”.
2. Type of forwarding that was invoked - is one of the following types. For each type the values "on" and "off" is allowed:

   Immediate - Forwarding all calls,

   Busy - Forwarding when busy,

   No Answer - Forwarding after no answer,

   Immediate Internal - Forwarding all internal calls,

   Immediate External - Forwarding all external calls,

   Busy Internal - Forwarding when busy for an internal call,

   Busy External - Forwarding when busy for an external call,

   No Answer Internal - Forwarding after no answer for an internal call,

   No Answer External- Forwarding after no answer for an external call.

The Event Report also may include one or more of the following parameters:

3. CSTA Device Identifier - indicates the device to which calls are forwarded.
4. CSTA Private Data Information.
5. CSTA Security Service Information.

Device.OnMessageWaiting Event

This Event Report indicates that the Message Waiting feature has been invoked for a device.

Note 8:

This Event Report can be provided to both monitors providing Event Reports for the call and monitors for the device to which the Message Waiting is directed. Specifically, it can be provided to both the subject and object devices of the feature.

The Message Waiting Event Report includes the following parameters:

1. CSTA Device Identifier - indicates the device at which Message Waiting was invoked. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
2. CSTA Device Identifier - indicates the device where the message is waiting. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
3. Flag - indicates whether the feature was turned on or off.

This Event Report also may include one or more of the following parameters:

4. CSTA Private Data Information.
5. CSTA Security Service Information.

Device.OnAutoAnswer Event

This Event Report indicates that the Auto-Answer feature has been changed for a device.

This Event Report includes the following parameters:

1. CSTA Device Identifier - indicates the device at which Auto-Answer was invoked. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
2. Flag - indicates whether the feature was turned on or off.

This Event Report also may include one or more of the following parameters:

3. CSTA Private Data Information.
4. CSTA Security Service Information.

Device.OnMicrophoneMute Event

This Event Report indicates that the Microphone Mute feature has been changed for a device.

This Event Report includes the following parameters:

1. CSTA Device Identifier - indicates the device for which Microphone Mute was invoked. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
2. Flag - indicates whether the microphone was turned on or off.

This Event Report also may include one or more of the following parameters:

3. CSTA Private Data Information.
4. CSTA Security Service Information.

Device.OnSpeakerMute Event

This Event Report indicates that the Speaker Mute feature has been changed for a device.

This Event Report includes the following parameters:

1. CSTA Device Identifier - indicates the device for which Speaker Mute was invoked. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
2. Flag - indicates whether the speaker was turned on or off.

This Event Report also may include one or more of the following parameters:

3. CSTA Private Data Information.
4. CSTA Security Service Information.

Device.OnSpeakerVolume Event

This Event Report indicates that the Speaker Volume has been changed for a device.

This Event Report includes the following parameters:

1. CSTA Device Identifier - indicates the device at which speaker volume was changed. If a device is not specified then this parameter indicates “Unknown” or “Not Required”.
2. Speaker Volume - indicates the volume that has been set. The value is an integer from 0 (silent) to 100 (maximum volume).

This Event Report also may include one or more of the following parameters:

3. CSTA Private Data Information.
4. CSTA Security Service Information.

Maintenance Event Reports

Each Maintenance Event Report is a message that indicates a change in the maintenance state of a device in the CSTA network. Each Maintenance Event Report indicates the new state that the device enters, independent of any previous state.

Every Event Report includes the identifier of the device to which the event applies. Additionally, every Event Report may include a cause value or field that clarifies the basic meaning of the Event Report. Every Event Report also may include CSTA Private Data Information.

System.OnBackInService Event

This Event Report indicates that the device has returned into service and once again operates normally in the CSTA domain.

The Back In Service Event Report includes the following parameter:

1. CSTA Device Identifier - indicates the device that is Back In Service. If a device is not specified then this parameter indicates “Not Required”.

The Event Report also may include one or more of the following parameters:

2. Cause - indicates a reason for the Event Report.
3. CSTA Private Data Information.
4. CSTA Security Service Information.

System.OnOutOfService Event

This Event Report indicates that the device has entered a maintenance state (i.e. has been taken Out Of Service) and can no longer accept calls or be manipulated via CSTA. It may be possible to continue to Monitor or take a Snapshot of such a device, but no direct Services (such as Make Call or Request Feature) can be provided.

The Out Of Service Event Report includes the following parameter:

1. CSTA Device Identifier - indicates the device that is taken Out Of Service. If a device is not specified then this parameter indicates “Not Required”.

The Event Report also may include one or more of the following parameters:

2. Cause - indicates a reason for the Event Report.
3. CSTA Private Data Information.
4. CSTA Security Service Information.

Prepared	Subject Responsible	Document Identity		
Mikica B Kocic	Mikica B Kocic	f92e8c8d-d3b0-47e6-b009-41acf714b5e9		
Approved / Checked		Date	Rev.	File
-		2002-03-12	43	-

Private Event Reports

System.OnPrivate Event

Private Event Reports may be used to convey proprietary event information to an application that cannot be reported within any of the defined Event Reports.

Voice Unit Event Reports

Each Voice Unit Event Report is a message that indicates a change in state of a message in a Voice Unit in the CSTA network. Voice Unit Event Reports indicates the new state that the message enters regardless of any previous state.

Every Event Report includes both the identifier of the message and that of the connection to the Voice Unit that is being monitored. Additionally, Event Reports may include the message length, the current position in the message, a cause for the event, and CSTA Private Data Information.

The model represented in Figure 9 illustrates the relationships among the following events.

Voice.OnPlay Event

This Event Report indicates that a message is being played.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the connection for the voice path to the Voice Unit for the affected message.
2. Message Identifier - indicates the message that changed state.

The Event Report also may include one or more of the following parameters:

3. Message Length - indicates the length of the message in milliseconds.
4. Current Position in Message - indicates the number of milliseconds from the start of the message.
5. Speed - indicates playing speed as percentage of nominal (normal) speed.
6. Cause - indicates a reason for the occurrence of the Event Report.
7. CSTA Private Data Information.
8. CSTA Security Service Information.

Voice.OnRecord Event

This Event Report indicates that a message is being recorded.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the connection for the voice path to the Voice Unit for the affected message.
2. Message Identifier - indicates the message that changed state.

The Event Report also may include one or more of the following parameters:

3. Message Length - indicates the length of the message in milliseconds.
4. Current Position in Message - indicates the number of milliseconds from the start of the message.
5. Cause - indicates a reason for the occurrence of the Event Report.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Voice.OnReview Event

This Event Report indicates that a recording message is being reviewed.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the connection for the voice path to the Voice Unit for the affected message.
2. Message Identifier - indicates the message that changed state.

The Event Report also may include one or more of the following parameters:

3. Message Length - indicates the length of the message in milliseconds.
4. Current Position in Message - indicates the number of milliseconds from the start of the message.
5. Cause - indicates a reason for the occurrence of the Event Report.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Voice.OnStop Event

This Event Report indicates that a message is not being used for play or record.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the connection for the voice path to the Voice Unit for the affected message.
2. Message Identifier - indicates the message that changed state.

The Event Report also may include one or more of the following parameters:

3. Message Length - indicates the length of the message in milliseconds.
4. Current Position in Message - indicates the number of milliseconds from the start of the message.
5. Cause - indicates a reason for the occurrence of the Event Report.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

End of Message Detected, Message Length Exceeded, and No Speech Detected.

Voice.OnSuspendPlay Event

This Event Report indicates that a message is suspended in play.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the connection for the voice path to the Voice Unit for the affected message.
2. Message Identifier - indicates the message that changed state.

The Event Report also may include one or more of the following parameters:

3. Message Length - indicates the length of the message in milliseconds.
4. Current Position in Message - indicates the number of milliseconds from the start of the message.
5. Cause - indicates a reason for the occurrence of the Event Report.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Voice.OnSuspendRecord Event

This Event Report indicates that a message is suspended during recording.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the connection for the voice path to the Voice Unit for the affected message.
2. Message Identifier - indicates the message that changed state.

The Event Report also may include one or more of the following parameters:

3. Message Length - indicates the length of the message in milliseconds.
4. Current Position in Message - indicates the number of milliseconds from the start of the message.
5. Cause - indicates a reason for the occurrence of the Event Report.
6. CSTA Private Data Information.
7. CSTA Security Service Information.

Frequently used cause codes are listed below. For the complete list see EventCauseType Enumeration.

Message Length Exceeded, and No Speech Detected.

Voice.OnVoiceAttributesChange Event

This Event Report indicates that one or more attributes of a message has changed.

The Event Report includes the following parameters:

1. CSTA Connection Identifier - indicates the connection for the voice path to the Voice Unit for the affected message.
2. Message Identifier - indicates the message for which one or more attributes changed.

The Event Report also may include one or more of the following parameters:

3. Volume - indicates the current volume on the specified connection. A value of 100 indicates maximum volume and 0 indicates minimum volume.
4. Level - indicates the current recording level on the specified connection. A value of 100 indicates maximum recording level and 0 indicates minimum level.
5. Speed - indicates current speed of the specified message. A value of 100% indicates normal speed. The slowest speed is 1%.
6. Current Position in Message - indicates the number of milliseconds from the start of the specified message.
7. Cause - indicates a reason for the occurrence of the Event Report.
8. CSTA Private Data Information.
9. CSTA Security Service Information.

Enumerations

ErrorStatus Enumeration

Hirerachy

CTNGLib → **ErrorStatus**

Description

ErrorStatus is returned by CTNG object's method that invokes a CSTA service.

Status OK Diagnostic

No-error status, i.e. status OK is represented by the following enumerated value:

Table 2 - ErrorStatus Values – No Error Value

Constant	Value	Description
CTNG_NO_ERROR	0x0000	Operation successfully completed.

Error Categories

Errors are grouped in classes (categories) with purpose to identify part of the CTNG subsystem that caused an error. Following error categories exist:

Table 3 - ErrorStatus Values – Error Classes

Constant	Value	Description
CSER_INTERNAL_ERROR	0x0000	CTNG internal service and communication errors.
CSER_REJECT _GENERAL_PROBLEM	0x0100	ROSE reject general problem error class.
CSER_REJECT _INVOKE_PROBLEM	0x0200	ROSE reject invoke problem error class.
CSER_REJECT _RETURN_RESULT_PROBLEM	0x0300	ROSE return result problem error class.
CSER_REJECT _RETURN_ERROR_PROBLEM	0x0400	ROSE reject return error problem class.
CSER_OPERATIONAL_ERROR	0x0500	CSTA operational error.
CSER_STATE_ERROR	0x0600	CSTA state error.
CSER_SYSTEM_RESOURCE_ERROR	0x0700	CSTA system resource error.
CSER_SUBSCRIBED_RESOURCE _AVAILABILITY_ERROR	0x0800	CSTA subscribed resource availability error.
CSER_PERFORMANCE_ERROR	0x0900	CSTA performance error class.
CSER_SECURITY_ERROR	0x0A00	CSTA security error class.
CSER_UNSPECIFIED_ERROR	0x0B00	CSTA unspecified error class.
CSER_NON_STANDARD_ERROR	0x0C00	CSTA non standard error class.

Internal and Communication Errors

Table 4 - ErrorStatus Values – Internal Errors

Constant	Value	Description
CSTA_ERROR_CODE_NOT_SET	0x0001	Error code is not set by internal engine.
CSTA_ERROR_TIMEOUT	0x0002	Connection timeout error.
CSTA_CONNECTION_NOT_OPEN	0x0003	Connection to PABX is not open.
CSTA_NOT_IMPLEMENTED	0x0004	CSTA service/feature is not implemented.

ROSE Diagnostic Errors

Table 5 - ErrorStatus Values – ROSE Reject Genral Problem Errors

Constant	Value	Description
RJGP_unrecognisedAPDU	0x0100	Unrecognized APDU
RJGP_mistypedAPDU	0x0101	Mistyped APDU
RJGP_badlyStructuredAPDU	0x0102	Badly structured APDU
RJIP_duplicateInvocation	0x0200	Duplicate Invoke ID

Table 6 - ErrorStatus Values – ROSE Reject Invoke Problem Errors

Constant	Value	Description
RJIP_unrecognisedOperation	0x0201	Unrecognized operation
RJIP_mistypedArgument	0x0202	Mistyped argument
RJIP_resourceLimitation	0x0203	Resource limitation
RJIP_initiatorReleasing	0x0204	Initiator is releasing invoke
RJIP_unrecognisedLinkedID	0x0205	Unrecognized
RJIP_linkedResponseUnexpected	0x0206	Linked response is not expected
RJIP_unexpectedChildOperation	0x0207	Child operation is not expected

Table 7 - ErrorStatus Values – ROSE Reject Return Result Problem Errors

Constant	Value	Description
RJRRP_unrecognisedInvocation	0x0300	Unrecognized invoke ID
RJRRP_resultResponseUnexpected	0x0301	Result response is not expected
RJRRP_mistypedResult	0x0302	Mistyped returned result

Table 8 - ErrorStatus Values – ROSE Reject Return Error Problem Errors

Constant	Value	Description
RJREP_unrecognisedInvocation	0x0400	Unrecognized invoke ID
RJREP_errorResponseUnexpected	0x0401	Error response is not expected
RJREP_unrecognisedError	0x0402	Unrecognized error
RJREP_unexpectedError	0x0403	Unexpected error
RJREP_mistypedParameter	0x0404	Mistyped error parameter

CSTA Diagnostic Errors

Error values available to return in response to a Service Request are those defined in this sub-clause. The definitions apply equally to Services requested by a CSTA Computing Function and to those requested by a CSTA Switching Function. An error value indicates the server's best evaluation of the condition that caused the server to send a negative response to the Service Request.

Each Error value comprises a mandatory error category indicator and an optional, specific error value. Errors are grouped into the following categories:

1. Operation Errors
2. Security Errors
3. State Incompatibility Errors
4. System Resource Availability Errors
5. Subscribed Resource Availability Errors
6. Performance Management Errors
7. Private Data Information Errors
8. Unspecified Errors

Operation errors

An error value in this category indicates an error in the Service Request. Each error value has the meaning ascribed to it in the following list. This category includes at least one of the following specific error values:

Table 9 - ErrorStatus Values – Universal Failure Operations Errors

Constant	Value	Description
ERUFOP_generic	0x0501	Generic Operation Error. The server has detected an error in the operation class, but it is not one of the defined errors, or the server cannot be any more specific.
ERUFOP_requestIncompatibleWithObject	0x0502	Request Incompatible With Object. The Request is not compatible with the object.
ERUFOP_valueOutOfRange	0x0503	Value Out Of Range. A parameter has a value that is not in the range defined for the server.
ERUFOP_objectNotKnown	0x0504	Object Not Known. A parameter has a value that is not known to the server.
ERUFOP_invalidCallingDevice	0x0505	Invalid Calling Device. The calling device is not valid.
ERUFOP_invalidCalledDevice	0x0506	Invalid Called Device. The called device is not valid.
ERUFOP_invalidForwardingDestination	0x0507	Privilege Violation on Specified Device. The specified device is not authorized for the requested Service.
ERUFOP_privilegeViolationOnSpecifiedDevice	0x0508	Invalid Forwarding Destination. The forwarding destination device is not valid.
ERUFOP_privilegeViolationOnCalledDevice	0x0509	Privilege Violation On Called Device. The called device is not authorized for the Service.
ERUFOP_privilegeViolationOnCallingDevice	0x050A	Privilege Violation On Calling Device. The calling device is not authorized for the Service.
ERUFOP_invalidCSTACallIdentifier	0x050B	Invalid CSTA Call Identifier. The Call Identifier is not valid.
ERUFOP_invalidCSTADeviceIdentifier	0x050C	Invalid CSTA Device Identifier. The Device Identifier is not valid.

ERUFOP_invalidCSTAConnectionIdentifier	0x050D	Invalid CSTA Connection Identifier. The Connection Identifier is not valid.
ERUFOP_invalidDestination	0x050E	Invalid Destination. The Service Request specified a destination that is not valid.
ERUFOP_invalidFeature	0x050F	Invalid Feature. The Service Request specified a feature that is not valid.
ERUFOP_invalidAllocationState	0x0510	Invalid Allocation State. The Service Request specified an allocation condition that was not valid.
ERUFOP_invalidCrossRefID	0x0511	Invalid CSTA Cross Reference Identifier. The Service Request specified a Cross Reference Identifier that is not in use at the time.
ERUFOP_invalidObjectType	0x0512	Invalid Object Type. The Service Request specified an object type that is outside the range of valid object types for the Service.
ERUFOP_securityViolation	0x0513	Security Violation. The requested Service would violate a security requirement.

Security errors

An error value in this category indicates a security error. Each error value has the meaning ascribed to it in the following list. This category includes at least one of the following specific error values:

Table 10 - ErrorStatus Values – Universal Failure Security Errors

Constant	Value	Description
ERUFSE_unspecified	0x0A00	Generic Security Error. This is a general-purpose value that can be used when the server is unable to be any more specific about the cause of the error.
ERUFSE_sequenceNumberViolated	0x0A01	Sequence Number Error. Indicates that the server has detected an error in the operation's Sequence Number.
ERUFSE_timeStampViolated	0x0A02	Time Stamp Error. Indicates that the server has detected an error in the operation's Time Stamp.
ERUFSE_pACViolated	0x0A03	PAC Error. Indicates that the server has detected an error in the operation's PAC.
ERUFSE_sealViolated	0x0A04	Seal Error. Indicates that the server has detected an error in the operation's Seal.

State incompatibility errors

An error value in this category indicates that the Service Request was not compatible with the condition of a related CSTA object. Each error value has the meaning ascribed to it in the following list. This category includes at least one of the following specific error values:

Table 11 - ErrorStatus Values – Universal Failure State incompatibility Errors

Constant	Value	Description
ERUFSI_generic	0x0601	Generic State Incompatibility Error. This is a general-purpose value that can be used when the server is unable to be any more specific about the cause of the error.
ERUFSI_invalidObjectState	0x0602	Incorrect Object State. The object is in the incorrect state for the Service. This error value may be used when the server cannot be any more specific.
ERUFSI_invalidConnectionID	0x0603	Invalid CSTA Connection Identifier For Active Call. The Connection Identifier specified in the Active Call parameter of the request was not in the correct state.
ERUFSI_noActiveCall	0x0604	No Active Call. The requested Service operates on an active call, but there was no active call.
ERUFSI_noHeldCall	0x0605	No Held Call. The requested Service operates on a held call, but the specified call was not in the Hold state.
ERUFSI_noCallToClear	0x0606	No Call To Clear. There was no call associated with the Connection Identifier of the Clear Call request.

Constant	Value	Description
ERUFSI_noConnectionToClear	0x0607	No Connection To Clear. There was no Connection for the Connection Identifier specified as Connection To Be Cleared.
ERUFSI_noCallToAnswer	0x0608	No Call To Answer. There was no call active for the Connection Identifier specified as Call To Be Answered.
ERUFSI_noCallToComplete	0x0609	No Call To Complete. There was no call active for the Connection Identifier specified as Call To Be Completed.
(CSTA Phase II)	0x060A	Not Able to Play. The specified message exists, but cannot be played.
(CSTA Phase II)	0x060B	Not Able to Resume. The specified message cannot be resumed.
(CSTA Phase II)	0x060C	End of Message. The message pointer is at the end of the message.
(CSTA Phase II)	0x060D	Beginning of Message. The message pointer is at the beginning of the message.
(CSTA Phase II)	0x060E	Message Suspended. The specified message is already suspended on the same Connection.

System resource availability errors

An error value in this category indicates that the Service Request could not be fulfilled because of a lack of system resources within the serving sub-domain. Each error value has the meaning ascribed to it in the following list. This category includes at least one of the following specific error values:

Table 12 - ErrorStatus Values – Universal Failure System Resource Availability Errors

Constant	Value	Description
ERUFSYRA_generic	0x0701	Generic System Resource Availability Error. This is a general-purpose value that can be used when the server is unable to be any more specific about the cause of the error.
ERUFSYRA_serviceBusy	0x0702	Service Busy. The Service is supported by the server, but was temporarily unavailable.
ERUFSYRA_resourceBusy	0x0703	Resource Busy. An internal resource is busy and temporarily unavailable.
ERUFSYRA_resourceOutOfService	0x0704	Resource Out Of Service. The Service requires a resource that is Out Of Service. A Service Request that encounters this condition could initiate system problem determination actions (e.g. notification of the network administrator).
ERUFSYRA_networkBusy	0x0705	Network Busy. The server sub-domain is busy.
ERUFSYRA_networkOutOfService	0x0706	Network Out Of Service. The server sub-domain is Out Of Service.
ERUFSYRA_overallMonitorLimitExceeded	0x0707	Overall Monitor Limit Exceeded. The requested Service would exceed the server's overall limit of monitors.
ERUFSYRA_conferenceMemberLimitExceeded	0x0708	Conference Member Limit Exceeded. The requested Service would exceed the server's limit on the number of members of a conference.

Subscribed resource availability errors

An error value in this category indicates that the Service Request could not be fulfilled because a required resource must be purchased or contracted by the client system. Each error value has the meaning ascribed to it in the following list. This category includes at least one of the following specific error values:

Table 13 - ErrorStatus Values – Universal Failure Subscribed Resource Availability Errors

Constant	Value	Description
ERUFSCRA_generic	0x0801	Generic Subscribed Resource Availability Error. This is a general-purpose value to be used when the server is unable to be any more specific about the cause of the error.
ERUFSCRA_objectMonitorLimitExceeded	0x0802	Generic Subscribed Resource Availability Error. This is a general-purpose value to be used when the server is unable to be any more specific about the cause of the error.
ERUFSCRA_externalTrunkLimitExceeded	0x0803	Object Monitor Limit Exceeded. The requested Service would exceed the server's limit of monitors for the specified object.
ERUFSCRA_outstandingRequestLimitExceeded	0x0804	Outstanding Requests Limit Exceeded. The limit of outstanding requests would be exceeded by this request.

Performance management errors

An error value in this category indicates that an error has been returned as a performance management mechanism. Each error value has the meaning ascribed to it in the following list. This category includes at least one of the following specific error values:

1. Generic Performance Management Error. This is a general-purpose value to be used when the server is unable to be any more specific about the cause of the error.
2. Performance Limit Exceeded. A performance limit has been exceeded.

Table 14 - ErrorStatus Values – Universal Failure Performance Management Errors

Constant	Value	Description
ERUFPM_generic	0x0901	Generic Performance Management Error. This is a general-purpose value to be used when the server is unable to be any more specific about the cause of the error.
ERUFPM_performanceLimitExceeded	0x0902	Performance Limit Exceeded. A performance limit has been exceeded.

CSTA private data information errors

An error value in this category indicates an error in the CSTA Private Data Information of the Service Request. The reason(s) why the Private Data Information is incorrect is not relevant to CSTA Standard. This category includes the following specific error value:

Table 15 - ErrorStatus Values – Universal Failure Non Standard Errors

Constant	Value	Description
ERUFNSE_unspecified	0x0C00	Unspecified Error

Unspecified errors

An error value in this category indicates that the error did not belong to any of the other error value categories. This category includes the following error value:

Table 16 - ErrorStatus Values – Universal Failure Unspecified Error

Constant	Value	Description
ERUFUE_unspecified	0x0B00	Unspecified Error

See Also

See1

Applies To: reference

Event Report messages are sent from server to client when a monitor request has been positively acknowledged and a CSTA-reportable event has occurred. For monitors on the Switching Function, Event Reports are sent from the Switching Function to the Computing Function.

SystemStatusType Enumeration

Hierarchy

CTNGLib → **SystemStatusType**

Description

SystemStatusType is parameter used by the Device.SystemStatus service, and it is received as a parameter of Device.OnSystemStatus event.

Table 17 - SystemStatusType Enumeration Values

Constant	Value	Description
ss_unknown	-1	Undefined system status type.
ss_initializing	0	**Initializing** - indicates that the system is initializing or restarting. This status indicates that a system is temporarily unable to respond to any requests. If provided, this status message is followed by an Enable status message to indicate that the Init process has completed.
ss_enabled	1	**Enabled** - indicates that Requests and Responses have been enabled. This usually occurs after a disruption or restart. This status cause is always sent after an Initializing cause has been sent and may be sent under other conditions. This status indicates that there are no outstanding monitor requests.
ss_normal	2	**Normal** - may be sent at any time and indicates that the status is normal. This status has no effect on other Services.
ss_messagesLost	3	**Messages Lost** - indicates that Requests and/or Responses, including Event Reports, may have been lost.
ss_disabled	4	**Disabled** - indicates that existing Monitor Requests have been disabled. Other Requests and Responses also may be disabled, but reject responses are always provided.
ss_overloadImminent	5	**Overload Imminent** - indicates that the receiver is requested to take initiative to shed load.
ss_overloadReached	6	**Overload Reached** - indicates that the requester may take initiative to shed load. This cause may be followed by Stop Monitor requests sent to the client and by rejections to additional Service Requests.
ss_overloadRelieved	7	**Overload Relieved** - indicates that the overload condition has passed.

See Also

See1

Applies To: reference

EventCauseType Enumeration

Hierarchy

CTNGLib → **EventCauseType**

Description

Cause codes can be used to clarify other information provided in both CSTA Event Reports and responses to Service Requests.

Cause codes may appear in any call Event Report where they make sense. The following list of cause codes definitions shows how cause codes modify the Event Reports to which they apply. Following the list of cause codes is a table showing the cause codes that are meaningful for each CSTA Event Report.

Agent Event Cause codes

Cause code	**indicates that:**
Forced Pause	The agent entered a periodic non-working condition. This usually occurs because of regulations that require agents to have a certain amount of time between handling successive ACD calls.

Table 18 - CSTA Agent Event Report - Cause Relationships

Cause \ Event	Agent Logged On	Agent Logged Off	Agent Ready	Agent Not Ready	Agent Busy	Agent Working After Call
Forced Pause				y		

Call Event Cause codes

Cause code	**indicates that:**
Active Monitor	An Active Monitor Feature has been invoked. This feature is typically used to allow intrusion with the ability to speak and listen by a supervisor into an agent call. The resultant call may be considered to be a conference. Therefore, cause code may be supplied with the Conferenced Event Report.
Alternate	The call is in the process of being interchanged with another call. This feature is typically used with single-line telephones where the human interface involves placing one call on hold and retrieving a held call or answers a waiting call with a single action.
Blocked	One party has disconnected from a call leaving one other party remaining with a local connection.
Busy	The call encountered a busy or unavailable device.
Call Back	Call Back has been invoked to complete a call that encountered busy or a no answer condition. This cause may occur in a Service Initiated Event Report to indicate that the caller is being prompted.
Call Cancelled	The call has been terminated before the associated device has gone on-hook.
Call Forward	The call has been redirected for any reason via a Call Forwarding feature.
Call Fd. - Immediate	The call has been redirected via a Call Forwarding feature set for all conditions.
Call Fd. - Busy	The call has been redirected via a Call Forwarding feature set for a busy endpoint.

Call Fd. - No Answer	The call has been redirected via a Call Forwarding feature set for an endpoint that does not answer.
Call Not Answered	The call was not answered because a timer elapsed.
Call Pickup	The call was redirected via a Call Pickup feature.
Camp On	A Camp On feature was invoked or has matured.
Consultation	A Consultation has been undertaken and/or is in progress.
Dest. Not Obtainable	The call could not reach the destination.
Distributed	The call was distributed by an ACD or hunt group.
Do Not Disturb	The call encountered a Do Not Disturb condition.
Entering Distribution	The call was delivered to a distribution mechanism (ACD).
Incompatible Destination	The call encountered an incompatible destination.
Invalid Account Code	The call has an invalid account code.
Key Operation	The reported Event occurred at a bridged or twin device.

Note 9:

Telephone numbers associated primarily with one device often appear also on a second device. One example is a secretary whose phone has mirrored or bridged lines of a boss's phone.

Lockout	The call encountered inter-digit time-out while dialling.
Maintenance	The call encountered a facility or endpoint in a maintenance condition.
Make Call	The event was provided as the result of a CSTA Make Call Service This cause may occur in a Service Initiated Event Report to indicate that the caller is being prompted.
Network Congestion	The call encountered a congested network. In some circumstances this cause code indicates that the user is listening to a special signal tone from a network. The tone may be accompanied by a voiced statement similar to "All circuits are busy...".
Network Not Obtainable	The call could not reach a destination network.
Network Signal	The event was provided as result of a network (trunk supervision / call progress) signal.
New Call	The call has not yet been redirected.
No Available Agents	The call could not access any agent.
Normal Clearing	The call or connection cleared in a normal way.
Number Changed	The called number has been changed to a new number.
Overflow	The call overflowed a queue, group, or target.
Override	The call resulted from use of an Override feature.
Park	The Event Report is associated with an action to park a call or retrieve a call from park.
Recall	The call is alerting due to a time-out associated with a feature that failed to complete or that anticipated further action from the user.
Redirected	The call has been redirected.
Reorder Tone	The call encountered a reorder condition. When this occurs, the network usually provides Reorder Tone to indicate that a request (call, feature, or supplementary service) was not recognizable. This condition typically results when a user dials a number that is not valid or attempts to obtain a service that is not enabled for that user or device.

Resources not Available	Resources were not available.
Silent Monitor	The event was caused by the invocation of a feature that allows a third party, such as an ACD agent supervisor, to join the call. The joining party can hear the entire conversation, but cannot be heard by either original party. The feature, sometimes called silent intrusion, may provide a tone to one or both parties to indicate that they are being monitored. This feature is not the same as a CSTA Monitor request. This cause code is not used to indicate that a CSTA Monitor has been initiated.
Single Step Conference	The event occurred through the use of Single Step Conference Call Service.
Single Step Transfer	The Transfer occurred in a single step.
Time-out	The reported event was generated because a trunk timer expired.
Transfer	A Transfer is in progress or has occurred.
Trunks Busy	The call encountered Trunks Busy.
Voice Unit Initiator	The reported event resulted from action by automated equipment (voice mail device, voice response unit, announcement) rather than from action by a human user.

Prepared	Subject Responsible	Document Identity		
Mikica B Kocic	Mikica B Kocic	f92e8c8d-d3b0-47e6-b009-41acf714b5e9		
Approved	Checked	Date	Rev.	File
-		2002-03-12	43	-

Table 19 - CSTA Call Event Report - Cause Relationships

Cause \ Event	Call Clr.	Conf	Con. Clr.	Delv.	Divrt	Est.	Fail	Held	Net. Rch.	Orig.	Q-ed	Retr.	Svc. Init.	Tran.
Active Monitor		y												
Alternate						y	y	y				y		
Blocked							y							
Busy							y				y			
Call Back	y		y	y						y	y		y	
Call Cancelled	y		y				y						y	
Call Forward				y	y		y	y	y		y			
Call Fd. - Immediate				y	y		y		y		y			
Call Fd. - Busy				y	y		y		y		y			
Call Fd. - No Answer				y	y		y	y	y		y			
Call Not Answered	y		y		y		y							
Call Pickup					y	y								
Camp On				y			y				y			
Consultation								y						
Dest. not Obtainable			y				y				y			
Distributed	y			y	y	y	y		y		y			y
Do Not Disturb			y		y		y				y			
Entering Distribution				y		y					y			
Incpt. Destination	y		y		y		y							
Invalid Account Code	y						y							
Key Operation	y	y	y	y	y	y	y	y	y	y	y	y	y	y
Lockout							y							
Maintenance	y						y							
Make Call										y			y	
Net Congestion							y				y			
Net Not Obtainable							y				y			
Network Signal	y	y	y	y	y	y	y	y			y			y
New Call		y		y		y				y				y
No Available Agents				y	y		y				y			
Normal Clearing	y		y											
Number Changed							y							
Overflow	y		y	y	y		y		y		y			
Override	y	y	y	y		y	y			y			y	
Park			y								y			
Recall		y		y	y	y	y	y			y	y		y
Redirected				y	y		y		y		y			y
Reorder Tone							y							
Resrcs. not Available	y		y				y		y		y			
Silent Monitor		y								y				
Single Step Conference		y				y								
Single Step Transfer														y
Timeout						y								
Transfer				y		y	y	y	y		y	y		y
Trunks Busy							y				y			
Voice Unit Initiator					y									y

Voice Unit Event Cause codes

Cause code	indicates that:
End of Data Detected	The end of data was detected during a Play or Review message operation. Indicating end-of-data, as opposed to end-of-message, allows the possibility of playing multiple messages.
Msg. Size Exceeded	The maximum permitted size of the message was detected during a Record message operation.
No Speech Detected	A period of silence was detected during a Record message operation (no speech).
DTMF Tone Detected	A DTMF tone was detected during message playback or recording.
Duration Exceeded	The maximum time duration for playback or recording of a message was exceeded.
Speech Detected	Speech (or non-silence) was detected during message playback.
Next Message	indicates the start of play of the next message within a sequence of messages.

Table 20 - CSTA Voice Unit Event Report - Cause Relationships

Cause \ Event	Stop	Play	Suspend/ Play	Record	Suspend/ Record	Review
End of Data Detected	y					
Message Size Exceeded	y				y	
No Speech Detected	y	y			y	
DTMF Tone Detected	y	y	y	y	y	y
Duration Exceeded	y					
Speech Detected	y	y	y	y		
Next Message		y				

I/O Service Cause codes

Although there are no Event Reports associated with I/O Services, the Send Data Service allows cause codes as an optional parameter. These cause codes are as follows:

Cause code	indicates that:
Termination char. recv'd	The specified termination character was received.
Character count reached	Specified number of characters were entered.
Timeout	Timeout occurred.
Switch-terminated	The switch terminated collection before other termination condition encountered.

Table 21 - CSTA I/O Services Event Report - Cause Relationships

Cause \ Event	Send Data	Stop Data	Send Broadcast Data	Send multicast Data	Suspend Data Path	Data Path Suspended	Resume Data Path	FastData
Termination char. rcv'd	y							
Character count reached	y							
Timeout	y							
Switch-terminated	y							

Event Cause Enumeration Values

Table 22 - EventCauseType Enumeration Values

Constant	Value	Description
ec_unknown	-1	Unknown event cause.
ec_activeMonitor	1	An Active Monitor Feature has occurred. This feature typically allows intrusion by a supervisor into an agent call with the ability to speak and listen. The resultant call can be considered as a conference so this cause code may be supplied with the Conferenced Event Report.
ec_alternate	2	The call is in the process of being exchanged. This feature is typically found on single-line telephones, where the human interface puts one call on hold and retrieves a held call or answers a waiting call in an atomic action.
ec_busy	3	The call encountered a busy tone or device.
ec_callBack	4	Call Back is a feature invoked (by a user or via CSTA) in an attempt to complete a call that has encountered a busy or no answer condition. As a result of invoking the feature, the failed call is cleared and the call can be considered as queued. The switch may subsequently automatically retry the call (normally when the called party next becomes free). Consequently, this cause code may appear in Event Reports related to the feature invocation (Call Cleared, Connection Cleared and Queued) or related to the subsequent, retried call (Service Initiated, Originated, Delivered, and Established).
ec_callCancelled	5	The user has terminated a call without going on-hook.
ec_callForwardAlways	6	The call has been redirected via a Call Forwarding feature set for general, unknown, or multiple conditions.
ec_callForwardBusy	7	The call has been redirected via a Call Forwarding feature set for a busy endpoint.
ec_callForwardNoAnswer	8	The call has been redirected via a Call Forwarding feature set for an endpoint that does not answer.
ec_callForward	9	The call has been redirected via a Call Forwarding feature set for general, unknown, or multiple conditions.
ec_callNotAnswered	10	The call was not answered because a timer has elapsed.
ec_callPickup	11	The call has been redirected via a Call Pickup feature.

Constant	Value	Description
ec_campOn	12	A Camp On feature has been invoked or has matured.
ec_destNotObtainable	13	The call could not obtain the destination.
ec_doNotDisturb	14	The call encountered a Do Not Disturb condition.
ec_incompatibleDestination	15	The call encountered an incompatible destination.
ec_invalidAccountCode	16	The call has an invalid account code.
ec_keyConference	17	Indicates that the Event Report occurred at a bridged or twin device. *NOTE: Telephone numbers associated primarily with one device often appear alsoon a second device. One example is a secretary who's phone has mirrored or bridged lines of a boss's phone.*
ec_lockout	18	The call encountered inter-digit timeout while dialling.
ec_maintenance	19	The call encountered a facility or endpoint in a maintenance condition.
ec_networkCongestion	20	The call encountered a congested network. In some circumstances this cause code indicates that the user is listening to a "No Circuit" Special Information Tone (SIT) from a network that is accompanied by a statement similar to "All circuits are busy...".
ec_networkNotObtainable	21	The call could not reach a destination network.
ec_newCall	22	The call has not yet been redirected.
ec_noAvailableAgents	23	The call could not access any agent.
ec_override	24	The call resulted because of an Override feature.
ec_park	25	Indicates that the Event Report is associated with an action to place a call to or retrieve a call from a parked position. Placing a call in a park position releases the call from the parking device, but retains the call in the Switching Function so that it can be connected to another (or the same) device by invoking the un-parking feature there.
ec_overflow	26	The call overflowed a queue, group, or target.
ec_recall	27	The call is alerting a device due to a time-out built into a feature that failed to complete or that anticipated further action from the user.
ec_redirected	28	The call has been redirected.
ec_reorderTone	29	The call encountered reorder - a tone provided by a network to indicate that the request (call, feature, or supplementary service) was not recognizable. This condition usually results when a user dials a number that is not valid or attempts to obtain a service that is not enabled for that user or device. In some circumstances this cause code indicates that the user is listening to a "Reorder" Special Information Tone (SIT) from a network that is accompanied by a statement similar to "The call did not go through as dialled...".
ec_resourcesNotAvailable	30	Resources were not available.
ec_silentParticipation	31	The event was caused by the invocation of a feature that allows a third party, such as an ACD agent supervisor, to join the call. The joining party can hear the entire conversation, but cannot be heard by either original party. The feature, sometimes called *silent intrusion*, may provide a tone to

Constant	Value	Description
		one or both parties to indicate that they are being monitored. This feature is not the same as a CSTA Monitor request. This cause does not indicate that a CSTA Monitor has been initiated.
ec_transfer	32	A Transfer is in progress or has occurred.
ec_trunksBusy	33	The call encountered Trunks Busy.
ec_voiceUnitInitiator	34	Indicates that the event was the result of action by automated equipment (voice mail device, voice response unit, announcement) rather than the result of action by a human user.

See Also

See1

Applies To: reference

LocalConnectionStateType Enumeration

Hierarchy

CTNGLib → **LocalConnectionStateType**

Description

One of a set of states a Connection may have. Connection states may be reported by Snapshots on either calls or devices, and changes in Connection states may be reported as Event Reports by Monitors. The Connection state refers to a single Call/Device relationship. A simplified Connection state model is given in Figure 6.

In Figure 6, the states (circles) presented are the CSTA set. The transitions between states, shown by arrows, show the typical states possible to enter from a given state and form the basis for providing Event Reports when they occur. These states are not equivalent to ISDN access states. They are a derivation of the state machine on one side of an ISDN access. The states are defined as follows:

Table 23 - LocalConnectionStateType Enumeration Values

Constant	Value	Description
lcs_unknown	-1	Unknown local connection state.
lcs_null	0	**Null** - the state where there is no relationship between the call and device.
lcs_initiated	1	**Initiated** - the state where the device is requesting service. Usually this results in the creation of a call. Often this is the "dialling" state.
lcs_alerting	2	**Alerting** - the state where a device is alerting (ringing). This indicates that a call wishes to become Connected to a device.
lcs_connected	3	**Connected** - the state where a device actively participates in a call. This state includes the notion of logical participation in a call as well as a physical participation in that call (i.e. Not Held).
lcs_hold	4	**Hold** - the state where a device inactively participates in a call. This state embodies the notion of logical participation in a call with suspended physical participation in that call.
lcs_queued	5	**Queued** - the state where normal state progression has been stalled. This state generally refers to two conditions but can apply to others as well. One condition is when a device is trying to establish a Connection with a call, and the process is stalled. The second condition is when a call tries to establish a Connection with a device and that process is stalled.
lcs_fail	6	**Fail** - the state where normal state progression has been aborted. This state generally refers to the condition when a device tries to become Connected to a call or a call tries to become Connected to a device and the attempt fails. Failed can result because of failure to connect the calling device and call, failure to connect the called device and call, failure to create the call, and other reasons.

See Also

See1

Applies To: reference

CTNG CSTA Phase I PICS

A.1 Introduction

The Protocol Implementation Conformance Statement (PICS) is a statement of which capabilities and options of the protocol have been implemented. The PICS can have a number of uses, including use:

- by the protocol implementer, as a check-list to reduce the risk of failure to conform to the standard through oversight;
- by the supplier and acquirer (or potential acquirer) of the implementation, as a detailed indication of the capabilities of the implementation, stated relative to the common basis for understanding provided by the standard PICS proforma;
- by the user (or potential user) of the implementation, as a basis for initially checking the possibility of interworking with another implementation (note that, while interworking cannot be guaranteed, failure to interwork can often be predicted from incompatible PICS);
- by a protocol tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

A.2 Definitions and abbreviations

CSTA Standard uses the following terms defined in ISO 9646-1:

- Protocol Implementation Conformance Statement (PICS);
- PICS Proforma.

In the "Reference" columns of the body of the PICS proforma, the letter S refers to the CSTA Services standard ECMA-179 and the letter P refers to the CSTA Protocol standard ECMA-180.

The following terms are used in the "Status" columns of the body of the PICS proforma:

m	=	**mandatory**; the capability is required for conformance to the protocol.
o	=	**optional**; the capability is not required for conformance to the protocol, or is required only = within constraints described in dependencies ("if" statements). If the capability is implemented, it is required to conform to the protocol specifications.
o.<n>	=	**optional**, but support of at least one of the group of options labelled by the same numeral <n> is required.
C.<cid>	=	**conditional**; the requirement is conditional according to the condition identified by <cid>.
<item>	=	simple-predicate condition, dependent on the support marked for <item>.

A.3 Conformance

The supplier of a protocol implementation which is claimed to conform to ECMA-180 shall complete a copy of the Protocol Implementation Conformance Statement (PICS) proforma in clauses A.5 to A.14.

A.4 Instructions for completing the PICS proforma

The first part of the PICS proforma, the Implementation Identification (clause A.5), is to be completed as indicated with the information necessary to identify fully both the supplier and the implementation.

The main part of the PICS proforma (clauses A.6 to A.14) is a fixed format questionnaire divided into subclauses each containing a group of individual items. Answers to the questionnaire items are to be provided in the rightmost column, either by marking an answer to indicate a restricted choice (usually Yes or No), or by checking off all supported values (for parameters with a default).

Each item is identified by an item reference in the first column; the second column title indicates the nature of the table items which follow. The third column contains the references to material that specifies the item in the main body of ECMA-179 and ECMA-180. The remaining columns record the status of the item - whether support is mandatory, optional, or not applicable - and provide space for the answers.

Where a service is not supported, any parameters or dependent service components are not applicable. These dependencies are indicated in the status column using the item identifier as a key. It is not necessary to complete items in any subsidiary sections if a "No" response is given to the primary service component.

For supported services, a negative response to a mandatory subsidiary item indicates that the service does not conform to ECMA-180, and conformance cannot be claimed for that service.

A.5 Implementation Identification

Supplier	**IPTC Technology Communication AB**
Protocol Version	**First Edition – Phase I**
Date of Statement	**2002-02-23**
Contact point for queries about the PICS	**http://www.iptc.se/doc/CTNGV10RM.pdf**
Implementation Name(s) and Version(s)	**CTNG™ v1.0**
Other information necessary for full identification - e.g. name(s) and version(s) for machines and/or operating systems; system name(s)	

Note A.1:

The first five items are required for all implementations; other information may be completed as appropriate in meeting the requirement for full identification.

Note A.2:

The terms Name and Version should be interpreted appropriately to correspond with a supplier's terminology (e.g. Type, Series, Model).

A.6 Switching Function Services

Alternate Call

Item	Service / Feature	Reference	Status	N/A	Supported?
A1	Alternate Call service	S9.1 P9.1	o		Yes [ ] No [ ]
A1a	Service Result	P5.3 P9.1	A1:m	[ ]	Yes [ ]
A1b	Held Call parameter	P9.1	A1:o1		Yes [ ] No [ ]
A1c	Active Call parameter	P9.1	A1:o1		Yes [ ] No [ ]
A1d	Security parameters	P16.8	A1:o		Yes [ ] No [ ]
A1e	Private Data in Request	P16.8	A1:o		Yes [ ] No [ ]
A1f	Private Data in Result	P16.8	A1:o		Yes [ ] No [ ]
A1g	Report of Service Errors	P14	A1:m	[ ]	Yes [ ]

Answer Call

Item	Service / Feature	Reference	Status	N/A	Supported?
A2	Answer Call service	S9.2 P9.2	o		Yes [ ] No [ ]
A2a	Service Result	P5.3 P9.2	A2:m	[ ]	Yes [ ]
A2b	Call to be Answered parameter	P9.2	A2:m	[ ]	Yes [ ]
A2c	Security parameters	P16.8	A2:o		Yes [ ] No [ ]
A2d	Private Data in Request	P16.8	A2:o		Yes [ ] No [ ]
A2e	Private Data in Result	P16.8	A2:o		Yes [ ] No [ ]
A2f	Report of Service Errors	P14	A2:m	[ ]	Yes [ ]

Call Completion

Item	Service / Feature	Reference	Status	N/A	Supported?
A3	Call Completion Service	S9.3 P9.3	o		Yes [ ] No [ ]
A3a	Camp On feature	P9.3	A3:o1		Yes [ ] No [ ]
A3b	Call Back feature	P9.3	A3:o1		Yes [ ] No [ ]
A3c	Intrude feature	P9.3	A3:o1		Yes [ ] No [ ]
A3d	Service Result	P5.3 P9.3	A3:m	[ ]	Yes [ ]
A3e	Call to Complete parameter	P9.3	A3:m	[ ]	Yes [ ]
A3f	Security parameters	P16.8	A3:o		Yes [ ] No [ ]
A3g	Private Data in Request	P16.8	A3:o		Yes [ ] No [ ]
A3h	Private Data in Result	P16.8	A3:o		Yes [ ] No [ ]
A3i	Report of Service Errors	P14	A3:m	[ ]	Yes [ ]

Clear Call

Item	Service / Feature	Reference	Status	N/A	Supported?

Item	Service / Feature	Reference	Status	N/A	Supported?
A4	Clear Call service	S9.4 P9.4	o		Yes [ ] No [ ]
A4a	Service Result	P5.3 P9.4	A4:m	[ ]	Yes [ ]
A4b	Call to be Cleared parameter	P9.4	A4:m	[ ]	Yes [ ]
A4c	Security parameters	P16.8	A4:o		Yes [ ] No [ ]
A4d	Private Data in Request	P16.8	A4:o		Yes [ ] No [ ]
A4e	Private Data in Result	P16.8	A4:o		Yes [ ] No [ ]
A4f	Report of Service Errors	P14	A4:m	[ ]	Yes [ ]

Clear Connection

Item	Service / Feature	Reference	Status	N/A	Supported?
A5	Clear Connection service	S9.5 P9.5	o		Yes [ ] No [ ]
A5a	Service Result	P5.3 P9.5	A5:m	[ ]	Yes [ ]
A5b	Connection to be Cleared parameter	P9.5	A5:m	[ ]	Yes [ ]
A5c	Security parameters	P16.8	A5:o		Yes [ ] No [ ]
A5d	Private Data in Request	P16.8	A5:o		Yes [ ] No [ ]
A5e	Private Data in Result	P16.8	A5:o		Yes [ ] No [ ]
A5f	Report of Service Errors	P14	A5:m	[ ]	Yes [ ]

Conference Call

Item	Service / Feature	Reference	Status	N/A	Supported?
A6	Conference Call service	S9.6 P9.6	o		Yes [ ] No [ ]
A6a	Service Result	P5.3 P9.6	A6:m	[ ]	Yes [ ]
A6b	Held Call parameter	P9.6	A6:o1		Yes [ ] No [ ]
A6c	Active Call parameter	P9.6	A6:o1		Yes [ ] No [ ]
A6d	Conference Call parameter in Result	P9.6	A6a:m	[ ]	Yes [ ]
A6e	Connection ID list	P9.6	A6:o		Yes [ ] No [ ]
A6f	Static Device ID included in list	P16.4	A6e:o	[ ]	Yes [ ] No [ ]
A6g	Security parameters	P16.8	A6:o		Yes [ ] No [ ]
A6h	Private Data in Request	P16.8	A6:o		Yes [ ] No [ ]
A6i	Private Data in Result	P16.8	A6:o		Yes [ ] No [ ]
A6j	Report of Service Errors	P14	A6:m	[ ]	Yes [ ]

Consultation Call

Item	Service / Feature	Reference	Status	N/A	Supported?
A7	Consultation Call service	S9.7 P9.7	o		Yes [ ] No [ ]
A7a	Service Result	P5.3 P9.7	A7:m	[ ]	Yes [ ]

Item	Service / Feature	Reference	Status	N/A	Supported?
A7b	Existing call parameter	P9.7	A7:m	[ ]	Yes [ ]
A7c	Called Device ID parameter	P9.7	A7:m	[ ]	Yes [ ]
A7d	Initiated Call parameters in Result	P9.7	A7a:m	[ ]	Yes [ ]
A7e	Security parameters	P16.8	A7:o		Yes [ ] No [ ]
A7f	Private Data in Request	P16.8	A7:o		Yes [ ] No [ ]
A7g	Private Data in Result	P16.8	A7:o		Yes [ ] No [ ]
A7h	Report of Service Errors	P14	A7:m	[ ]	Yes [ ]

Divert Call

Item	Service / Feature	Reference	Status	N/A	Supported?
A8	Divert Call service	S9.8 P9.8	o		Yes [ ] No [ ]
A8a	Deflect feature	P9.8	A8:o1		Yes [ ] No [ ]
A8b	Directed Pickup feature	P9.8	A8:o1		Yes [ ] No [ ]
A8c	Group Pickup feature	P9.8	A8:o1		Yes [ ] No [ ]
A8d	Service Result	P5.3 P9.8	A8:m	[ ]	Yes [ ]
A8e	Call to be Diverted parameter	P16.6	A8a:m	[ ]	Yes [ ]
A8f	New Destination parameter	P16.6	A8a:m	[ ]	Yes [ ]
A8g	Call to be Picked Up parameter	P16.6	A8b:m	[ ]	Yes [ ]
A8h	Requesting Device parameter	P16.6	A8b:o	[ ]	Yes [ ]
A8i	Device ID of group parameter	P16.6	A8c:m	[ ]	Yes [ ]
A8j	Security parameters	P16.8	A8:o		Yes [ ] No [ ]
A8k	Private Data in Request	P16.8	A8:o		Yes [ ] No [ ]
A8l	Private Data in Result	P16.8	A8:o		Yes [ ] No [ ]
A8m	Report of Service Errors	P14	A8:m	[ ]	Yes [ ]

Hold Call

Item	Service / Feature	Reference	Status	N/A	Supported?
A9	Hold Call service	S9.9 P9.9	o		Yes [ ] No [ ]
A9a	Service Result	P5.3 P9.9	A9:m	[ ]	Yes [ ]
A9b	Call to be Held	P9.9	A9:m	[ ]	Yes [ ]
A9c	Procedures for Connection Reservation parameter = TRUE	S9.9.1	A9:o		Yes [ ] No [ ]
A9d	Security parameters	P16.8	A9:o		Yes [ ] No [ ]
A9e	Private Data in Request	P16.8	A9:o		Yes [ ] No [ ]
A9f	Private Data in Result	P16.8	A9:o		Yes [ ] No [ ]
A9g	Repor of Service Errors	P14	A9:m	[ ]	Yes [ ]

Make Call

Item	Service / Feature	Reference	Status	N/A	Supported?
A10	Make Call service	S9.10 P9.10	o		Yes [ ] No [ ]
A10a	Service Result	P5.3 P9.10	A10:m	[ ]	Yes [ ]
A10b	Calling Device ID parameter	P9.10	A10:m	[ ]	Yes [ ]
A10c	Called Device ID parameter	P9.10	A10:m	[ ]	Yes [ ]
A10d	Initiated Call parameter in Result	P9.10	A10a:m	[ ]	Yes [ ]
A10e	Security parameters	P16.8	A10:o		Yes [ ] No [ ]
A10f	Private Data in Request	P16.8	A10:o		Yes [ ] No [ ]
A10g	Private Data in Result	P16.8	A10:o		Yes [ ] No [ ]
A10h	Report of Service Errors	P14	A10:m	[ ]	Yes [ ]

Make Predictive Call

Item	**Service / Feature**	**Reference**	**Status**	**N/A**	**Supported?**
A11	Make Predictive service	S9.11 P9.11	o		Yes [ ] No [ ]
A11a	Service Result	P5.3 P9.11	A11:m	[ ]	Yes [ ]
A11b	Calling Device ID parameter	P9.11	A11:m	[ ]	Yes [ ] No [ ]
A11c	Called Device ID parameter	P9.11	A11:m	[ ]	Yes [ ] No [ ]
A11d	Allocation on Established condition	S9.11.1	A11:o		Yes [ ] No [ ]
A11e	Initiated Call parameter in Result	P9.11	A11a:m	[ ]	Yes [ ]
A11f	Security parameters	P16.8	A11:o		Yes [ ] No [ ]
A11g	Private Data in Request	P16.8	A11:o		Yes [ ] No [ ]
A11h	Private Data in Result	P16.8	A11:o		Yes [ ] No [ ]
A11i	Report of Service Errors	P14	A11:m	[ ]	Yes [ ]

Query Device

Item	Service / Feature	Reference	Status	N/A	Supported?
A12	Query Device service	S9.12 P9.12	o		Yes [ ] No [ ]
A12a	Message Waiting feature	P9.12	A12:o1		Yes [ ] No [ ]
A12b	Do Not Disturb feature	P9.12	A12:o1		Yes [ ] No [ ]
A12c	Forwarding feature	P9.12	A12:o1		Yes [ ] No [ ]
A12d	Last Number feature	P9.12	A12:o1		Yes [ ] No [ ]
A12e	Device Info feature	P9.12	A12:o1		Yes [ ] No [ ]
A12f	Agent State feature	P9.12	A12:o1		Yes [ ] No [ ]
A12g	Service Result	P5.3 P9.12	A12:m	[ ]	Yes [ ]
A12h	Device parameter in Request	P9.12	A12:m	[ ]	Yes [ ]
A12i	Feature parameter in Request	P9.12	A12:m	[ ]	Yes [ ]
A12j	Device Information in Service Result	P9.12	A12g:m	[ ]	Yes [ ]

Item	Service / Feature	Reference	Status	N/A	Supported?
A12k	Message Waiting On	P16.6	A12a:m	[ ]	Yes [ ] No [ ]
A12l	Do Not Disturb On	P16.6	A12b:m	[ ]	Yes [ ] No [ ]
A12m	Forward Immediate	P16.6	A12c:o1	[ ]	Yes [ ] No [ ]
A12n	Forward Busy	P16.6	A12c:o1	[ ]	Yes [ ] No [ ]
A12o	Forward No Answer	P16.6	A12c:o1	[ ]	Yes [ ] No [ ]
A12p	Forward Busy Internal	P16.6	A12c:o1	[ ]	Yes [ ] No [ ]
A12q	Forward Busy External	P16.6	A12c:o1	[ ]	Yes [ ] No [ ]
A12r	Forward No Answer Internal	P16.6	A12c:o1	[ ]	Yes [ ] No [ ]
A12s	Forward No Answer External	P16.6	A12c:o1	[ ]	Yes [ ] No [ ]
A12t	Forward-to Number	P16.6	A12c:m	[ ]	Yes [ ]
A12u	Last Dialled Number	P16.6	A12d:m	[ ]	Yes [ ]
A12v	Device ID	P16.6	A12e:o1	[ ]	Yes [ ] No [ ]
A12w	Device Type	P16.6	A12e:o1	[ ]	Yes [ ] No [ ]
A12x	Device Class	P16.6	A12e:o1	[ ]	Yes [ ] No [ ]
A12y	Null	P16.6	A12f:o1	[ ]	Yes [ ] No [ ]
A12z	Not Ready	P16.6	A12f:o1	[ ]	Yes [ ] No [ ]
A12aa	Ready	P16.6	A12f:o1	[ ]	Yes [ ] No [ ]
A12bb	Work Not Ready	P16.6	A12f:o1	[ ]	Yes [ ] No [ ]
A12cc	Work Ready	P16.6	A12f:o1	[ ]	Yes [ ] No [ ]
A12dd	Security parameters	P16.8	A12:o		Yes [ ] No [ ]
A12ee	Private Data in Request	P16.8	A12:o		Yes [ ] No [ ]
A12ff	Private Data in Result	P16.8	A12:o		Yes [ ] No [ ]
A12gg	Report of Service Errors	P14	A12:m	[ ]	Yes [ ]

Reconnect Call

Item	Service / Feature	Reference	Status	N/A	Supported?
A13	Reconnect Call service	S9.13 P9.13	o		Yes [ ] No [ ]
A13a	Service Result	P5.3 P9.13	A13:m	[ ]	Yes [ ]
A13b	Held Call parameter	P9.13	A13:01		Yes [ ] No [ ]
A13c	Active Call parameter	P9.13	A13:01		Yes [ ] No [ ]
A13d	Security parameters	P16.8	A13:o		Yes [ ] No [ ]
A13e	Private Data in Request	P16.8	A13:o		Yes [ ] No [ ]
A13f	Private Data in Result	P16.8	A13:o		Yes [ ] No [ ]
A13g	Report of Service Errors	P14	A13:m	[ ]	Yes [ ]

Retrieve Call

Item	Service / Feature	Reference	Status	N/A	Supported?

Item	Service / Feature	Reference	Status	N/A	Supported?
A14	Retrive Call service	S9.14 P9.14	o		Yes [ ] No [ ]
A14a	Service Result	P5.3 P9.14	A14:m	[ ]	Yes [ ]
A14b	Call to be Retrieved parameter	P9.14	A14:m	[ ]	Yes [ ]
A14c	Security parameters	P16.8	A14:o		Yes [ ] No [ ]
A14d	Private Data in Request	P16.8	A14:o		Yes [ ] No [ ]
A14e	Private Data in Result	P16.8	A14:o		Yes [ ] No [ ]
A14f	Report of Service Errors	P14	A14:m	[ ]	Yes [ ]

Set Feature

Item	Service / Feature	Reference	Status	N/A	Supported?
A15	Set Feature service	S9.15 P9.15	o		Yes [ ] No [ ]
A15a	Message Waiting feature	P9.15	A15:o1		Yes [ ] No [ ]
A15b	Do Not Disturb feature	P9.15	A15:o1		Yes [ ] No [ ]
A15c	Forwarding feature	P9.15	A15:o1		Yes [ ] No [ ]
A15d	Agent Parameter feature	P9.15	A15:o1		Yes [ ] No [ ]
A15e	Service Result	P5.3 P9.15	A15:m	[ ]	Yes [ ]
A15f	Device parameter in Request	P9.15	A15:m	[ ]	Yes [ ]
A15g	Feature parameter in Request	P9.15	A15:m	[ ]	Yes [ ]
A15h	Forward Always	P9.15	A15c:01	[ ]	Yes [ ] No [ ]
A15i	Forward Busy	P9.15	A15c:o1	[ ]	Yes [ ] No [ ]
A15j	Forward No Answer	P9.15	A15c:o1	[ ]	Yes [ ] No [ ]
A15k	Forward Busy Internal	P9.15	A15c:o1	[ ]	Yes [ ] No [ ]
A15l	Forward Busy External	P9.15	A15c:o1	[ ]	Yes [ ] No [ ]
A15m	Forward No Answer Internal	P9.15	A15c:o1	[ ]	Yes [ ] No [ ]
A15n	Forward No Answer External	P9.15	A15c:o1	[ ]	Yes [ ] No [ ]
A15o	Forward to Device	P9.15	A15c:o	[ ]	Yes [ ] No [ ]
A15p	Login	P9.15	A15d:o1	[ ]	Yes [ ] No [ ]
A15q	Logout	P9.15	A15d:o1	[ ]	Yes [ ] No [ ]
A15r	Ready	P9.15	A15d:o1	[ ]	Yes [ ] No [ ]
A15s	Not Ready	P9.15	A15d:o1	[ ]	Yes [ ] No [ ]
A15t	Work Not Ready	P9.15	A15d:01	[ ]	Yes [ ] No [ ]
A15u	Work Ready	P9.15	A15d:o1	[ ]	Yes [ ] No [ ]
A15v	Agent ID	P9.15	c1:o	[ ]	Yes [ ] No [ ]
A15w	ACD Pilot or Group	P9.15	c1:o	[ ]	Yes [ ] No [ ]
A15x	Agent Password	P9.15	A15q:o	[ ]	Yes [ ] No [ ]
A15y	Security parameters	P16.8	A15:o		Yes [ ] No [ ]

Set Feature (continued)

Item	Service / Feature	Reference	Status	N/A	Supported?
A15z	Private Data in Request	P16.8	A15:o		Yes [ ] No [ ]
A12aa	Private Data in Result	P16.8	A15:o		Yes [ ] No [ ]
A12bb	Report of Service Errors	P14	A15:m	[ ]	Yes [ ]

Note A.3:

c1: (A15p or A15q)

Transfer Call

Item	Service / Feature	Reference	Status	N/A	Supported?
A16	Transfer Call service	S9.16 P9.16	o		Yes [ ] No [ ]
A16a	Service Result	P5.3 P9.16	A16:m	[ ]	Yes [ ]
A16b	Held Call parameter	P9.16	A16:m	[ ]	Yes [ ]
A16c	Active Call parameter	P9.16	A16:m	[ ]	Yes [ ]
A16d	Transferred Call parameter in Result	P9.16	A16:o		Yes [ ] No [ ]
A16e	List of remaining parties	P9.16	A16:o		Yes [ ] No [ ]
A16f	Static IDs included in list	P16.4	A16e:o	[ ]	Yes [ ] No [ ]
A16g	Security parameters	P16.8	A16:o		Yes [ ] No [ ]
A16h	Private Data in Request	P16.8	A16:o		Yes [ ] No [ ]
A16i	Private Data in Result	P16.8	A16:o		Yes [ ] No [ ]
A16j	Report of Service Errors	P14	A16:m	[ ]	Yes [ ]

A.7 Switching Function Events

Call Events

Event Macro

Item	Service / Feature	Reference	Status	N/A	Supported?
B1	Event macro	S10.2 P10	o		Yes [ ] No [ ]
B1a	Monitor CrossRefID	S10.2.1 P5.4 P10	B1:m	[ ]	Yes [ ]
B1b	EventTypeID	P10	B1:m	[ ]	Yes [ ]
B1c	EventInfo	P10	B1:m	[ ]	Yes [ ]
B1d	CSTA Private Data	P10	B1:o		Yes [ ] No [ ]

Call Cleared

Item	Service / Feature	Reference	Status	N/A	Supported?
B2	Call Cleared event	S10.2.3.1 P10.1.1	o		Yes [ ] No [ ]
B2a	Cleared Call parameter	P10.1.1	B2:m	[ ]	Yes [ ]
B2b	Cause parameter	P10.1.1	B2:o		Yes [ ] No [ ]
B2c	Local Connection information	P10.1.1	B2:o		Yes [ ] No [ ]

Conferenced

Item	Service / Feature	Reference	Status	N/A	Supported?
B3	Conferenced event	S10.2.3.2 P10.1.2	o		Yes [ ] No [ ]
B3a	Primary Old Call parameter	P10.1.2	B3:m	[ ]	Yes [ ]
B3b	Secondary Old Call parameter	P10.1.2	B3:C.1	[ ]	Yes [ ] No [ ]
B3c	Conference Controller parameter	P10.1.2	B3:m	[ ]	Yes [ ]
B3d	Added Party parameter	P10.1.2	B3:m	[ ]	Yes [ ]
B3e	Cause parameter	P10.1.2	B3:o		Yes [ ] No [ ]
B3f	Local Connection information	P10.1.2	B3:o		Yes [ ] No [ ]
B3g	Connection ID list	P10.1.2	B3:o		Yes [ ] No [ ]
B3h	Static Device ID included in list	P16.4	B3g:o	[ ]	Yes [ ] No [ ]

C.1: If provided in previous events then mandatory, else optional

Connection Cleared

Item	Service / Feature	Reference	Status	N/A	Supported?
B4	Connection Cleared event	S10.2.3.3 P10.1.3	o		Yes [ ] No [ ]

Item	Service / Feature	Reference	Status	N/A	Supported?
B4a	Droppped Connection parameter	P10.1.3	B4:m	[ ]	Yes [ ]
B4b	Releasing Device parameter	P10.1.3	B4:m	[ ]	Yes [ ]
B4c	Cause parameter	P10.1.3	B4:o		Yes [ ] No [ ]
B4d	Local Connection information	P10.1.3	B4:o		Yes [ ] No [ ]

Delivered

Item	Service / Feature	Reference	Status	N/A	Supported?
B5	Delivered event	S10.2.3.4 P10.1.4	o		Yes [ ] No [ ]
B5a	Alerting Connection ID parameter	P10.1.4	B5:m	[ ]	Yes [ ]
B5b	Alerting Device ID parameter	P10.1.4	B5:m	[ ]	Yes [ ]
B5c	Calling Device ID parameter	P10.1.4	B5:m	[ ]	Yes [ ]
B5d	Called Device parameter	P10.1.4	B5:m	[ ]	Yes [ ]
B5e	Last Redirection Device parameter	P10.1.4	B5:m	[ ]	Yes [ ]
B5f	Cause parameter	P10.1.4	B5:o		Yes [ ] No [ ]
B5g	Local Connection information	P10.1.4	B5:o		Yes [ ] No [ ]

Diverted

Item	Service / Feature	Reference	Status	N/A	Supported?
B6	Diverted event	S10.2.3.5 P10.1.5	o		Yes [ ] No [ ]
B6a	Diverted Connection ID parameter	P10.1.5	B6:C.2	[ ]	Yes [ ] No [ ]
B6b	Diverting Device parameter	P10.1.5	B6:m	[ ]	Yes [ ]
B6c	New Destination parameter	P10.1.5	B6:m	[ ]	Yes [ ]
B6d	Cause parameter	P10.1.5	B6:o		Yes [ ] No [ ]
B6e	Local Connection information	P10.1.5	B6:o		Yes [ ] No [ ]

C.2: If the call alerted the device then mandatory, else optional

Established

Item	Service / Feature	Reference	Status	N/A	Supported?
B7	Established event	S10.2.3.6 P10.1.6	o		Yes [ ] No [ ]
B7a	Established Connection parameter	P10.1.6	B7:m	[ ]	Yes [ ]
B7b	Answering Device parameter	P10.1.6	B7:m	[ ]	Yes [ ]
B7c	Calling Device parameter	P10.1.6	B7:m	[ ]	Yes [ ]
B7d	Called Device parameter	P10.1.6	B7:m	[ ]	Yes [ ]
B7e	Last Redirection Device parameter	P10.1.6	B7:m	[ ]	Yes [ ]

Item	Service / Feature	Reference	Status	N/A	Supported?
B7f	Cause parameter	P10.1.6	B7:o		Yes [ ] No [ ]
B7g	Local Connection information	P10.1.6	B7:o		Yes [ ] No [ ]

Failed

Item	Service / Feature	Reference	Status	N/A	Supported?
B8	Failed event	S10.2.3.7 P10.1.7	o		Yes [ ] No [ ]
B8a	Failed Connection	P10.1.7	B8:m	[ ]	Yes [ ]
B8b	Failing Device parameter	P10.1.7	B8:m	[ ]	Yes [ ]
B8c	Called Device parameter	P10.1.7	B8:m	[ ]	Yes [ ]
B8d	Cause parameter	P10.1.7	B8:o		Yes [ ] No [ ]
B8e	Local Connection information	P10.1.7	B8:o		Yes [ ] No [ ]

Held

Item	Service / Feature	Reference	Status	N/A	Supported?
B9	Held event	S10.2.3.8 P10.1.8	o		Yes [ ] No [ ]
B9a	Held Connection	P10.1.8	B9:m	[ ]	Yes [ ]
B9b	Holding Device parameter	P10.1.8	B9:m	[ ]	Yes [ ]
B9c	Cause parameter	P10.1.8	B9:o		Yes [ ] No [ ]
B9d	Local Connection information	P10.1.8	B9:o		Yes [ ] No [ ]

Network Reached

Item	Service / Feature	Reference	Status	N/A	Supported?
B10	Network Reached event	S10.2.3.9 P10.1.9	o		Yes [ ] No [ ]
B10a	Connection ID parameter	P10.1.9	B10:m	[ ]	Yes [ ]
B10b	Trunk Used parameter	P10.1.9	B10:m	[ ]	Yes [ ]
B10c	Called Device parameter	P10.1.9	B10:m	[ ]	Yes [ ]
B10d	Cause parameter	P10.1.9	B10:o		Yes [ ] No [ ]
B10e	Local Connection information	P10.1.9	B10:o		Yes [ ] No [ ]

Originated

Item	Service / Feature	Reference	Status	N/A	Supported?
B11	Originated event	S10.2.3.10 P10.1.10	o		Yes [ ] No [ ]
B11a	Originated Connection parameter	P10.1.10	B11:m	[ ]	Yes [ ]
B11b	Calling Device parameter	P10.1.10	B11:m	[ ]	Yes [ ]

Item	Service / Feature	Reference	Status	N/A	Supported?
B11c	Called Device parameter	P10.1.10	B11:m	[ ]	Yes [ ]
B11d	Cause parameter	P10.1.10	B11:o		Yes [ ] No [ ]
B11e	Local Connection information	P10.1.10	B11:o		Yes [ ] No [ ]

Queued

Item	Service / Feature	Reference	Status	N/A	Supported?
B12	Queued event	S10.2.3.11 P10.1.11	o		Yes [ ] No [ ]
B12a	Queued Connection parameter	P10.1.11	B12:m	[ ]	Yes [ ]
B12b	Queue parameter	P10.1.11	B12:m	[ ]	Yes [ ]
B12c	Calling Device parameter	P10.1.11	B12:m	[ ]	Yes [ ]
B12d	Called Device parameter	P10.1.11	B12:m	[ ]	Yes [ ]
B12e	Last Redirection Device parameter	P10.1.11	B12:m	[ ]	Yes [ ]
B12f	Number of Calls in Queue	P10.1.11	B12:o		Yes [ ] No [ ]
B12g	Cause parameter	P10.1.11	B12:o		Yes [ ] No [ ]
B12h	Local Connection information	P10.1.11	B12:o		Yes [ ] No [ ]

Retrieved

Item	Service / Feature	Reference	Status	N/A	Supported?
B13	Retrieved event	S10.2.3.12 P10.1.12	o		Yes [ ] No [ ]
B13a	Retrieved Connection parameter	P10.1.12	B13:m	[ ]	Yes [ ]
B13b	Retrieving Device parameter	P10.1.12	B13:m	[ ]	Yes [ ]
B13c	Cause parameter	P10.1.12	B13:o		Yes [ ] No [ ]
B13d	Local Connection information	P10.1.12	B13:o		Yes [ ] No [ ]

Service Initiated

Item	Service / Feature	Reference	Status	N/A	Supported?
B14	Service Initiated event	S10.2.3.13 P10.1.13	o		Yes [ ] No [ ]
B14a	Initiated Connection parameter	P10.1.13	B14:m	[ ]	Yes [ ]
B14b	Cause parameter	P10.1.13	B14:o		Yes [ ] No [ ]
B14c	Local Connection information	P10.1.13	B14:o		Yes [ ] No [ ]

Transferred

Item	Service / Feature	Reference	Status	N/A	Supported?
B15	Transferred event	S10.2.3.14 P10.1.14	o		Yes [ ] No [ ]

Item	Service / Feature	Reference	Status	N/A	Supported?
B15a	Primary Old Call parameter	P10.1.14	B15:m	[ ]	Yes [ ]
B15b	Secondary Old Call parameter	P10.1.14	B15:C.3	[ ]	Yes [ ] No [ ]
B15c	Transferring Device parameter	P10.1.14	B15:m	[ ]	Yes [ ]
B15d	Transferred-to Device parameter	P10.1.14	B15:m	[ ]	Yes [ ]
B15e	Cause parameter	P10.1.14	B15:o		Yes [ ] No [ ]
B15f	Local Connection information	P10.1.14	B15:o		Yes [ ] No [ ]
B15g	Connection ID list	P10.1.14	B15:o		Yes [ ] No [ ]
B15h	Static Device ID included in list	P16.4	B15g:o	[ ]	Yes [ ] No [ ]

C.3: If parameter in previous events then mandatory, else optional

Agent Events

Logged On

Item	Service / Feature	Reference	Status	N/A	Supported?
B16	Logged On event	S10.2.2.1 P10.3.1	o		Yes [ ] No [ ]
B16a	Agent Device parameter	P10.3.1	B16:m	[ ]	Yes [ ]
B16b	Agent ID parameter	P10.3.1	B16:o		Yes [ ] No [ ]
B16c	Agent Group parameter	P10.3.1	B16:o		Yes [ ] No [ ]
B16d	Password parameter	P10.3.1	B16:o		Yes [ ] No [ ]

Logged Off

Item	Service / Feature	Reference	Status	N/A	Supported?
B17	Logged Off event	S10.2.2.2 P10.3.2	o		Yes [ ] No [ ]
B17a	Agent Device parameter	P10.3.2	B17:m	[ ]	Yes [ ]
B17b	Agent ID parameter	P10.3.2	B17:o		Yes [ ] No [ ]
B17c	Agent Group parameter	P10.3.2	B17:o		Yes [ ] No [ ]

Ready

Item	Service / Feature	Reference	Status	N/A	Supported?
B18	Ready event	S10.2.2.4 P10.3.4	o		Yes [ ] No [ ]
B18a	Agent Device parameter	P10.3.4	B18:m	[ ]	Yes [ ]
B18b	Agent ID parameter	P10.3.4	B18:o		Yes [ ] No [ ]

Not Ready

Item	Service / Feature	Reference	Status	N/A	Supported?
B19	Not Ready event	S10.2.2.3 P10.3.3	o		Yes [ ] No [ ]
B19a	Agent Device parameter	P10.3.3	B19:m	[ ]	Yes [ ]
B19b	Agent ID parameter	P10.3.3	B19:o		Yes [ ] No [ ]

Work Ready

Item	Service / Feature	Reference	Status	N/A	Supported?
B20	Work Ready event	P10.2.2.6 P10.3.6	o		Yes [ ] No [ ]
B20a	Agent Device parameter	P10.3.6	B20:m	[ ]	Yes [ ]
B20b	Agent ID parameter	P10.3.6	B20:o		Yes [ ] No [ ]

Work Not Ready

Item	Service / Feature	Reference	Status	N/A	Supported?
B21	Work Not Ready event	P10.2.2.5 P10.3.5	o		Yes [ ] No [ ]
B21a	Agent Device parameter	P10.3.5	B21:m	[ ]	Yes [ ]
B21b	Agent ID parameter	P10.3.5	B21:o		Yes [ ] No [ ]

Other Feature Events

Call Information

Item	Service / Feature	Reference	Status	N/A	Supported?
B22	Call Information event	S10.2.4.1 P10.2.1	o		Yes [ ] No [ ]
B22a	Connection ID parameter	P10.2.1	B22:m	[ ]	Yes [ ]
B22b	Device parameter	P10.2.1	B22:o		Yes [ ] No [ ]
B22c	Account Information	P10.2.1	B22:o		Yes [ ] No [ ]
B22d	Authorisation Code	P10.2.1	B22:o		Yes [ ] No [ ]

Do Not Disturb

Item	Service / Feature	Reference	Status	N/A	Supported?
B23	Do Not Disturb event	S10.2.4.2 P10.2.2	o		Yes [ ] No [ ]
B23a	Device parameter	P10.2.2	B23:m	[ ]	Yes [ ]
B23b	Do Not Disturb On	P10.2.2	B23:m	[ ]	Yes [ ]

Forwarding

Item	Service / Feature	Reference	Status	N/A	Supported?
B24	Forwarding event	S10.2.4.3 P10.2.3	o		Yes [ ] No [ ]
B24a	Device parameter	P10.2.3	B24:m	[ ]	Yes [ ]
B24b	Forwarding Information	P10.2.3	B24:m	[ ]	Yes [ ]
B24c	Forwarding Type parameter	P16.6	B24b:m	[ ]	Yes [ ]
B24d	Forward Immediate On	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24e	Forward Immediate Off	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24f	Forward Busy On	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24g	Forward Busy Off	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24h	Forward No Answer On	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24i	Forward No Answer Off	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24j	Forward Busy Internal On	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24k	Forward Busy Internal Off	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24l	Forward Busy External On	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24m	Forward Busy External Off	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24n	Fwd No Answer Internal On	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24o	Fwd No Answer Internal Off	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24p	Fwd No Answer External On	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24q	Fwd No Answer External Off	P16.6	B24c:o	[ ]	Yes [ ] No [ ]
B24r	Forward DN	P16.6	B24b:o		Yes [ ] No [ ]

Message Waiting

Item	Service / Feature	Reference	Status	N/A	Supported?
B25	Message Waiting event	S10.2.4.4 P10.2.4	o		Yes [ ] No [ ]
B25a	Device for Message	P10.2.4	B25:m	[ ]	Yes [ ]
B25b	Invoking Device	P10.2.4	B25:m	[ ]	Yes [ ]
B25c	Message Waiting On parameter	P10.2.4	B25:m	[ ]	Yes [ ]

Maintenance Events

Back In Service

Item	Service / Feature	Reference	Status	N/A	Supported?
B26	Back in Service event	S10.2.5.1 P10.4.1	o		Yes [ ] No [ ]
B26a	Device ID parameter	P10.4.1	B26:m	[ ]	Yes [ ]
B26b	Cause parameter	P10.4.1	B26:o		Yes [ ] No [ ]

Out of Service

Item	Service / Feature	Reference	Status	N/A	Supported?
B27	Out of Service event	S10.2.5.2 P10.4.2	o		Yes [ ] No [ ]
B27a	Device ID parameter	P10.4.2	B27:m	[ ]	Yes [ ]
B27b	Cause parameter	P10.4.2	B27:o		Yes [ ] No [ ]

Private Events

Item	Service / Feature	Reference	Status	N/A	Supported?
B28	Is/Are Private Event(s)	S10.2.1 P10.5	o		Yes [ ] No [ ]

A.8 Computing Services

Route Request

Item	Service / Feature	Reference	Status	N/A	Supported?
C1	Route Request service	S11.3 P11.1	o		Yes [ ] No [ ]
C1a	Cross Reference parameter	P11.1	C1:m	[ ]	Yes [ ]
C1b	Current Route parameter	P11.1	C1:m	[ ]	Yes [ ]
C1c	Calling Device parameter	P11.1	C1:o		Yes [ ] No [ ]
C1d	Routed Call parameter	P11.1	C1:o		Yes [ ] No [ ]
C1e	Route Select Algorithm	P11.1	C1:o		Yes [ ] No [ ]
C1f	Priority parameter	P11.1	C1:o		Yes [ ] No [ ]
C1g	Set-up information	P11.1	C1:o		Yes [ ] No [ ]
C1h	Security parameters	P16.8	C1:o		Yes [ ] No [ ]
C1i	Private Data in Request	P16.8	C1:o		Yes [ ] No [ ]
C1j	Report of Service Errors	P14	C1:m	[ ]	Yes [ ]

Re-Route Service

Item	Service / Feature	Reference	Status	N/A	Supported?
C2	Re-Route service	S11.1 P11.2	o		Yes [ ] No [ ]
C2a	Cross Reference parameter	P11.2	C2:m	[ ]	Yes [ ]
C2b	Security parameters	P16.8	C2:o		Yes [ ] No [ ]
C2c	Private Data in Request	P16.8	C2:o		Yes [ ] No [ ]
C2d	Report of Service Errors	P14	C2:m	[ ]	Yes [ ]

Route Select

Item	Service / Feature	Reference	Status	N/A	Supported?
C3	Route Select service	S11.4 P11.3	o		Yes [ ] No [ ]
C3a	Cross Reference parameter	P11.3	C3:m	[ ]	Yes [ ]
C3b	Route Selected parameter	P11.3	C3:m	[ ]	Yes [ ]
C3c	ISDN Set-Up parameter	P11.3	C3:o		Yes [ ] No [ ]
C3d	Remaining Retries parameter	P11.3	C3:o		Yes [ ] No [ ]
C3e	Route Used Request parameter	P11.3	C3:o		Yes [ ] No [ ]
C3f	Security parameters	P16.8	C3:o		Yes [ ] No [ ]
C3g	Private Data in Request	P16.8	C3:o		Yes [ ] No [ ]
C3h	Report of Service Errors	P14	C3:m	[ ]	Yes [ ]

Route Used

Item	Service / Feature	Reference	Status	N/A	Supported?

Item	Service / Feature	Reference	Status	N/A	Supported?
C4	Route Used service	S11.5 P11.4	o		Yes [ ] No [ ]
C4a	Cross Reference parameter	P11.4	C4:m	[ ]	Yes [ ]
C4b	Route Used parameter	P11.4	C4:m	[ ]	Yes [ ]
C4c	Calling Device parameter	P11.4	C4:o		Yes [ ] No [ ]
C4d	Domain parameter	P11.4	C4:o		Yes [ ] No [ ]
C4e	Security parameters	P16.8	C4:o		Yes [ ] No [ ]
C4f	Private Data in Request	P16.8	C4:o		Yes [ ] No [ ]
C4g	Report of Service Errors	P14	C4:m	[ ]	Yes [ ]

Route End

Item	Service / Feature	Reference	Status	N/A	Supported?
C5	Route End service	S11.2 P11.5	o		Yes [ ] No [ ]
C5a	Cross Reference parameter	P11.5	C5:m	[ ]	Yes [ ]
C5b	Error Value parameter	P11.5	C5:o		Yes [ ] No [ ]
C5c	Security parameters	P16.8	C5:o		Yes [ ] No [ ]
C5d	Private Data in Request	P16.8	C5:o		Yes [ ] No [ ]
C5e	Report of Service Errors	P14	C5:m	[ ]	Yes [ ]

A.9 Bidirectional Services

Escape Service

Item	Service / Feature	Reference	Status	N/A	Supported?
D1	Escape service	S12.1 P12.1	o		Yes [ ] No [ ]
D1a	Service Result	P5.3 P12.1	D1:m	[ ]	Yes [ ]
D1b	Security parameters	P16.8	D1:o		Yes [ ] No [ ]
D1c	Private Data in Request	P16.8	D1:o		Yes [ ] No [ ]
D1d	Private Data in Result	P16.8	D1:o		Yes [ ] No [ ]
D1e	Report of Service Errors	P14	D1:m	[ ]	Yes [ ]

System Status

Item	Service / Feature	Reference	Status	N/A	Supported?
D2	System Status service	S12.2 P12.2	o		Yes [ ] No [ ]
D2a	Service Result	P5.3 P12.2	D2:m	[ ]	Yes [ ]
D2b	System Status Cause	P12.2	D2:m	[ ]	Yes [ ]
D2c	Initializing	P12.2	D2:o		Yes [ ] No [ ]
D2d	Enabled	P12.2	D2:o		Yes [ ] No [ ]
D2e	Normal	P12.2	D2:o		Yes [ ] No [ ]
D2f	Messages Lost	P12.2	D2:o		Yes [ ] No [ ]
D2g	Disabled	P12.2	D2:o		Yes [ ] No [ ]
D2h	Overload Imminent	P12.2	D2:o		Yes [ ] No [ ]
D2i	Overload Reached	P12.2	D2:o		Yes [ ] No [ ]
D2j	Overload Relieved	P12.2	D2:o		Yes [ ] No [ ]
D2k	Security parameters	P16.8	D2:o		Yes [ ] No [ ]
D2l	Private Data in Request	P16.8	D2:o		Yes [ ] No [ ]
D2m	Private Data in Result	P16.8	D2:o		Yes [ ] No [ ]
D2n	Report of Service Errors	P14	D2:m	[ ]	Yes [ ]

A.10 Status Reporting Services

Change Monitor Filter

Item	Service / Feature	Reference	Status	N/A	Supported?
E1	Change Monitor Filter service	S10.1 P13.2	o		Yes [ ] No [ ]
E1a	Service Result	P5.3 P13.2	E1:m	[ ]	Yes [ ]
E1b	Cross Reference ID parameter in Request	P13.2	E1:m	[ ]	Yes [ ]
E1c	Filter List in Request	P13.2	E1:m	[ ]	Yes [ ]
E1d	Filter List in Result	P13.2	E1:o		Yes [ ] No [ ]
E1e	Security parameters	P16.8	E1:o		Yes [ ] No [ ]
E1f	Private Data in Request	P16.8	E1:o		Yes [ ] No [ ]
E1g	Private Data in Result	P16.8	E1:o		Yes [ ] No [ ]
E1h	Report of Service Errors	P14	E1:m	[ ]	Yes [ ]

Monitor Start

Item	Service / Feature	Reference	Status	N/A	Supported?
E2	Monitor Start	S10.3 P13.1	o		Yes [ ] No [ ]
E2a	Service Result	P5.3 P13.1	E2:m	[ ]	Yes [ ]
E2b	Monitor Object parameter	P13.1	E2:m	[ ]	Yes [ ]
E2c	Monitor Object device	P13.1	E2:o1		Yes [ ] No [ ]
E2d	Monitor Object call	P13.1	E2:o1		Yes [ ] No [ ]
E2e	Monitor Type parameter	P13.1	E2:o		Yes [ ] No [ ]
E2f	Monitor Type device	P13.1	c1:o1		Yes [ ] No [ ]
E2g	Monitor Type call	P13.1	c1:o1		Yes [ ] No [ ]
E2h	Monitor Type default to Call Monitoring	P13.1	E2:o1		Yes [ ] No [ ]
E2i	Monitor Type default to Device Monitoring	P13.1	E2:o1		Yes [ ] No [ ]
E2j	Monitor Filter parameter for Call Processing events	P13.1	E2:o		Yes [ ] No [ ]
E2k	Monitor Filter parameter for Feature events	P13.1	E2:o		Yes [ ] No [ ]
E2l	Monitor Filter parameter for Agent events	P13.1	E2:o		Yes [ ] No [ ]
E2m	Monitor Filter parameter for Maintenance events	P13.1	E2:o		Yes [ ] No [ ]
E2n	Cross Reference ID in Service Result	P13.1	E2a:m	[ ]	Yes [ ]
E2o	Security parameters	P16.8	E2:o		Yes [ ] No [ ]
E2p	Private Data in Request	P16.8	E2:o		Yes [ ] No [ ]

Item	Service / Feature	Reference	Status	N/A	Supported?
E2q	Private Data in Result	P16.8	E2:o		Yes [ ] No [ ]
E2r	Report of Service Errors	P14	E2:m	[ ]	Yes [ ]

Note A.4:

c1: (E2c or E2d)

Monitor Stop

Item	Service / Feature	Reference	Status	N/A	Supported?
E3	Monitor Stop	S10.4 P13.3	o		Yes [ ] No [ ]
E3a	Service Result	P5.3 P13.3	E3:m	[ ]	Yes [ ]
E3b	Cross Reference ID parameter in Request	P13.3	E3:m	[ ]	Yes [ ]
E3c	Security parameters	P16.8	E3:o		Yes [ ] No [ ]
E3d	Private Data in Request	P16.8	E3:o		Yes [ ] No [ ]
E3e	Private Data in Result	P16.8	E3:o		Yes [ ] No [ ]
E3f	Report of Service Errors	P14	E3:m	[ ]	Yes [ ]

Snapshot Call

Item	Service / Feature	Reference	Status	N/A	Supported?
E4	Snapshot Call service	S10.5 P13.5	o		Yes [ ] No [ ]
E4a	Service Result	P5.3 P13.5	E4:m	[ ]	Yes [ ]
E4b	Snapshot Object parameter in Request	P13.5	E4:m	[ ]	Yes [ ]
E4c	Static Device ID parameter in Result	P13.5	E4a:m	[ ]	Yes [ ]
E4d	Connection ID parameter in Result	P13.5	E4a:m	[ ]	Yes [ ]
E4e	Local Connection State parameter in Result	P13.5	E4a:o		Yes [ ] No [ ]
E4f	Security parameters	P16.8	E4:o		Yes [ ] No [ ]
E4g	Private Data in Request	P16.8	E4:o		Yes [ ] No [ ]
E4h	Private Data in Result	P16.8	E4:o		Yes [ ] No [ ]
E4i	Report of Service Errors	P14	E4:m	[ ]	Yes [ ]

Snapshot Device

Item	Service / Feature	Reference	Status	N/A	Supported?
E5	Snapshot Device service	S10.6 P13.4	o		Yes [ ] No [ ]
E5a	Service Result	P5.3 P13.4	E5:m	[ ]	Yes [ ]
E5b	Static Device ID parameter in Request	P13.4	E5a:m	[ ]	Yes [ ]

Item	Service / Feature	Reference	Status	N/A	Supported?
E5c	Connection ID parameter in Result	P13.4	E5a:m	[ ]	Yes [ ]
E5d	Call State parameter in Result	P13.4	E5a:m	[ ]	Yes [ ]
E5e	Security parameters	P16.8	E5:o		Yes [ ] No [ ]
E5f	Private Data in Request	P16.8	E5:o		Yes [ ] No [ ]
E5g	Private Data in Result	P16.8	E5:o		Yes [ ] No [ ]
E5h	Report of Service Errors	P14	E5:m	[ ]	Yes [ ]

A.11 Switching Event Cause Values

Item	Cause	Reference	Status	N/A	Supported?
F1	Cause values in event reports	S10.2.8 P15	o		Yes [ ] No [ ]
F1a	Active Monitor	P15	F1:o		Yes [ ] No [ ]
F1b	Alternate	P15	F1:o		Yes [ ] No [ ]
F1c	Busy	P15	F1:o		Yes [ ] No [ ]
F1d	Call Back	P15	F1:o		Yes [ ] No [ ]
F1e	Call Cancelled	P15	F1:o		Yes [ ] No [ ]
F1f	Call Forward Immediate	P15	F1:o		Yes [ ] No [ ]
F1g	Call Forward Busy	P15	F1:o		Yes [ ] No [ ]
F1h	Call Forward No Answer	P15	F1:o		Yes [ ] No [ ]
F1i	Call Forward	P15	F1:o		Yes [ ] No [ ]
F1j	Call Not Answered	P15	F1:o		Yes [ ] No [ ]
F1k	Call Pickup	P15	F1:o		Yes [ ] No [ ]
F1l	Camp On	P15	F1:o		Yes [ ] No [ ]
F1m	Destination Not Obtainable	P15	F1:o		Yes [ ] No [ ]
F1n	Do Not Disturb	P15	F1:o		Yes [ ] No [ ]
F1o	Incompatible Destination	P15	F1:o		Yes [ ] No [ ]
F1p	Invalid Account Code	P15	F1:o		Yes [ ] No [ ]
F1q	Key Operation	P15	F1:o		Yes [ ] No [ ]
F1r	Lockout	P15	F1:o		Yes [ ] No [ ]
F1s	Maintenance	P15	F1:o		Yes [ ] No [ ]
F1t	Network Congestion	P15	F1:o		Yes [ ] No [ ]
F1u	Network Not Obtainable	P15	F1:o		Yes [ ] No [ ]
F1v	New Call	P15	F1:o		Yes [ ] No [ ]
F1w	No Available Agents	P15	F1:o		Yes [ ] No [ ]
F1x	Overflow	P15	F1:o		Yes [ ] No [ ]
F1y	Override	P15	F1:o		Yes [ ] No [ ]
F1z	Park	P15	F1:o		Yes [ ] No [ ]
F1aa	Recall	P15	F1:o		Yes [ ] No [ ]
F1ab	Redirected	P15	F1:o		Yes [ ] No [ ]
F1ac	Reorder Tone	P15	F1:o		Yes [ ] No [ ]
F1ad	Resources Not Available	P15	F1:o		Yes [ ] No [ ]
F1ae	Silent Monitor	P15	F1:o		Yes [ ] No [ ]
F1af	Transfer	P15	F1:o		Yes [ ] No [ ]
F1ag	Trunks Busy	P15	F1:o		Yes [ ] No [ ]
F1ah	Voice Unit Initiator	P15	F1:o		Yes [ ] No [ ]

A.12 Switching Function Errors

Operational Errors

Item	Error	Reference	Status	N/A	Supported?
G1	Operational Errors category	S8.4.1 P14	o		Yes [ ] No [ ]
G1a	Generic	P14	G1:o		Yes [ ] No [ ]
G1b	Request Incompatible with Object	P14	G1:o		Yes [ ] No [ ]
G1c	Value Out of Range	P14	G1:o		Yes [ ] No [ ]
G1d	Object Not Known	P14	G1:o		Yes [ ] No [ ]
G1e	Invalid Calling Device	P14	G1:o		Yes [ ] No [ ]
G1f	Invalid Called Device	P14	G1:o		Yes [ ] No [ ]
G1g	Privilege Violation on Specified Device	P14	G1:o		Yes [ ] No [ ]
G1h	Invalid Forwarding Destination	P14	G1:o		Yes [ ] No [ ]
G1i	Privilege Violation on Called Device	P14	G1:o		Yes [ ] No [ ]
G1j	Privilege Violation on Calling Device	P14	G1:o		Yes [ ] No [ ]
G1k	Invalid Call Identifier	P14	G1:o		Yes [ ] No [ ]
G1l	Invalid Device Identifier	P14	G1:o		Yes [ ] No [ ]
G1m	Invalid Connection Identifier	P14	G1:o		Yes [ ] No [ ]
G1n	Invalid Destination	P14	G1:o		Yes [ ] No [ ]
G1o	Invalide Feature	P14	G1:o		Yes [ ] No [ ]
G1p	Invalid Allocation State	P14	G1:o		Yes [ ] No [ ]
G1q	Invalid Cross Reference ID	P14	G1:o		Yes [ ] No [ ]
G1r	Invalid Object Type	P14	G1:o		Yes [ ] No [ ]
G1s	Security Violation	P14	G1:o		Yes [ ] No [ ]

State Incompatibility

Item	Error	Reference	Status	N/A	Supported?
G2	State Incompatibility category	S8.4.3 P14	o		Yes [ ] No [ ]
G2a	Generic	P14	G2:o		Yes [ ] No [ ]
G2b	Incorrect Object State	P14	G2:o		Yes [ ] No [ ]
G2c	Invalid Connection ID	P14	G2:o		Yes [ ] No [ ]
G2d	No Active Call	P14	G2:o		Yes [ ] No [ ]
G2e	No Held Call	P14	G2:o		Yes [ ] No [ ]
G2f	No Call to Clear	P14	G2:o		Yes [ ] No [ ]
G2g	No Connection to Clear	P14	G2:o		Yes [ ] No [ ]
G2h	No Call to Answer	P14	G2:o		Yes [ ] No [ ]
G2i	No Call to Complete	P14	G2:o		Yes [ ] No [ ]

System Resource Availability Errors

Item	Error	Reference	Status	N/A	Supported?
G3	System Resource Availability Errors category	S8.4.4 P14	o		Yes [ ] No [ ]
G3a	Generic	P14	G3:o		Yes [ ] No [ ]
G3b	Service Busy	P14	G3:o		Yes [ ] No [ ]
G3c	Resource Busy	P14	G3:o		Yes [ ] No [ ]
G3d	Resource Out of Service	P14	G3:o		Yes [ ] No [ ]
G3e	Network Busy	P14	G3:o		Yes [ ] No [ ]
G3f	Network Out of Service	P14	G3:o		Yes [ ] No [ ]
G3g	Overall Monitor Limit Exceeded	P14	G3:o		Yes [ ] No [ ]
G3h	Conference Member Limit Exceeded	P14	G3:o		Yes [ ] No [ ]

Subscribed Resource Availability Errors

Item	Error	Reference	Status	N/A	Supported?
G4	Subscribed Resource Availability Errors category	S8.4.5 P14	o		Yes [ ] No [ ]
G4a	Generic	P14	G4:o		Yes [ ] No [ ]
G4b	Object Monitor Limit Exceeded	P14	G4:o		Yes [ ] No [ ]
G4c	External Trunck Limit Exceeded	P14	G4:o		Yes [ ] No [ ]
G4d	Outstanding Request Limit Exceeded	P14	G4:o		Yes [ ] No [ ]

Performance Errors

Item	Error	Reference	Status	N/A	Supported?
G5	Performance Errors category	S8.4.6 P14	o		Yes [ ] No [ ]
G5a	Generic	P14	G5:o		Yes [ ] No [ ]
G5b	Performance Limit Exceeded	P14	G5:o		Yes [ ] No [ ]

A.13 CSTA Data Types

Item	Data Type	Reference	Status	N/A	Supported?
H1	Dynamic Device IDs	S6.1.1 P16.3	o		Yes [ ] No [ ]
H2	Extended Device IDs (choices follow)	S6.1.1 P16.2	m	[ ]	Yes [ ]
H2a	Device Identifier	P16.2	H2:o1		Yes [ ] No [ ]
H2b	Implicit Public	P16.2	H2:o1		Yes [ ] No [ ]
H2c	Explicit Public	P16.2	H2:o1		Yes [ ] No [ ]
H2d	Implicit Private	P16.2	H2:o1		Yes [ ] No [ ]
H2e	Explicit Private	P16.2	H2:o1		Yes [ ] No [ ]
H2f	Other plan	P16.2	H2:o1		Yes [ ] No [ ]
H3	Device IDs (choices follow)	S6.1.1 P16.2	m	[ ]	Yes [ ]
H3a	Number digits	P16.2	H3:o1		Yes [ ] No [ ]
H3b	Device number	P16.2	H3:o1		Yes [ ] No [ ]

A.14 Security

Item	Service / Feature	Reference	Status	N/A	Supported?
I1	Security option	S7 P7 P16.7	o		Yes [ ] No [ ]
I1a	Message Sequence Number	P16.7	I1:o		Yes [ ] No [ ]
I1b	Time Stamp	P16.7	I1:o		Yes [ ] No [ ]
I1c	Privilege Attribute Certificate (PAC)	P16.7	I1:o		Yes [ ] No [ ]
I1d	Seal	P16.7	I1:o		Yes [ ] No [ ]