Standard ECMA-269
3rd Edition - December 1998

ECMA

Standardizing Information and Communication Systems

Services for Computer Supported Telecommunications Applications (CSTA) Phase III

Volume 4

Phone: +41 22 849.60.00 - **Fax:** +41 22 849.60.01 - **URL:**http://www.ecma.ch - **Internet:** helpdesk@ecma.ch

Standard ECMA-269
3rd Edition - December 1998

ECMA

Standardizing Information and Communication Systems

Services for Computer Supported Telecommunications Applications (CSTA) Phase III

Volume 4

Phone: +41 22 849.60.00 - **Fax:** +41 22 849.60.01 - **URL:**http://www.ecma.ch - **Internet:** helpdesk@ecma.ch

Brief History

This Standard ECMA-269 defines Phase III of Services for Computer Supported Telecommunications Applications (CSTA) for OSI Layer 7 communication between a computing network and a telecommunications network. This Standard is part of a Suite of Standards and Technical Reports for Phase III of CSTA. All of the Standards and Technical Reports in the Suite are based on practical experience of ECMA member companies and each one represents a pragmatic and widely-based consensus.

The evolution of this Suite began with CSTA Phase I, which included only the CSTA Services and Protocol Standards (ECMA-179 and ECMA-180). In Phase II, Technical Report ECMA TR/68 was added illustrating how CSTA services and events may be used in typical call scenarios. That Technical Report reflected a common understanding of ECMA member companies.

Phase III of CSTA extends the previous Phase II Standards (ECMA-217 and ECMA-218) in major theme directions as well as numerous details. This incorporates technology based upon the *versit* CTI Encyclopedia (Version 1.0), which was contributed to ECMA by *versit*. Major areas of advancement include:

- New categories of services and events such as capabilities exchange, charging, media attachment services, call data recording (CDR), etc.
- Additional services and events for call and device control.
- Enhancement to existing services and events.
- Organization of services and events to reflect a grouping based on function (call control, device control, etc.).
- Use of a consistent template for services and events that includes initial/final connection state, connection state transitions, event monitoring sequences, etc.

The First Edition of Standard ECMA-269 was published in December 1997 and the Second Edition was published in June 1998.

This edition completes the planned Services for CSTA Phase III by extending the Second Edition in the following areas: ACD and ACD Agent Modeling, Call Associated Features, Call Detail Recording services, Capability Exchange services, Data Collection services, I/O Services, Logical Device Feature services, Physical Device Feature services, Media Attachment services, Maintenance events, Vendor Specific Extensions, and Voice services.

This ECMA Standard is contributed to ISO/IEC JTC1 under the terms of the fast-track procedure, for adoption as an ISO/IEC International Standard.

Adopted as 3rd Edition of Standard ECMA-269 by the General Assembly of December 1998.

Table of Contents

Annex A
(normative)

Device Appearances

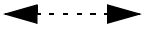
This annex describes the types of appearances supported by CSTA devices and their associated behaviour. For an introduction to logical elements and appearances, refer to 6.1.3.2, “Logical Element”, on page 14.

The type of appearance is based on its relationship with other devices. The type of appearance plays a great role in determining the functionality and behaviour associated with the logical element of the device. There are two types of appearances:

- *Standard* Appearance (see A.1)
- *Bridged* Appearance (see A.2)

The following notation is used in the figures throughout this Annex.

D	represents another device
L	represents the identifiers for the logical elements
A	represents an appearance of a logical element



indicates that there is an interaction and/or association between the elements or components of an element.

A.1 Standard Appearance

A standard appearance of a device’s logical element is not associated with other devices (i.e., cannot handle calls at another device). When a call arrives at the logical element, standard appearances are selected according to their behaviour:

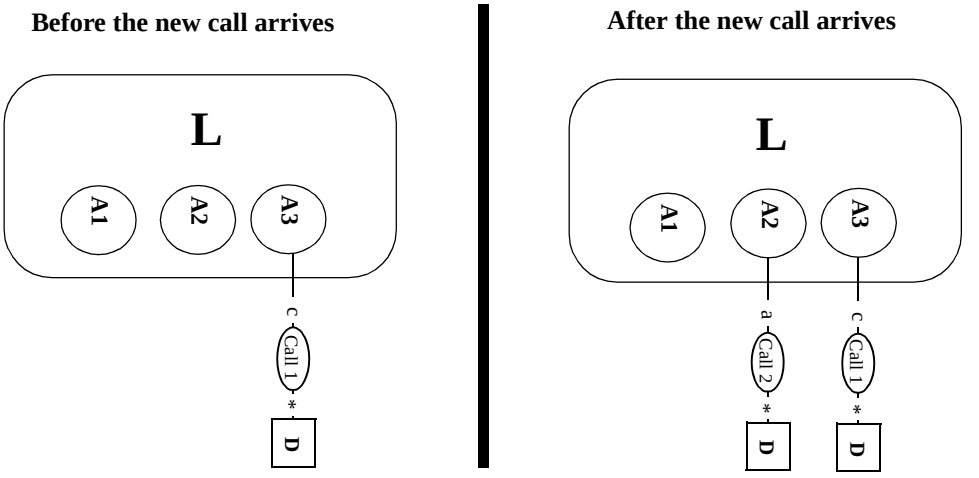
- Selected-Standard Appearance (see A.1.1)
- Basic-Standard Appearance (see A.2.1)

In addition, all media streams associated with calls corresponding to the appearances are only available to the physical element of the device (if it has one). This type of appearance can be either addressable or non-addressable. Refer to Clause 10, “CSTA Device Identifier Formats”, on page 75 for information on how to reference this appearance. When a call activity at one appearance results in the need for another appearance (i.e., the appearance puts the call on hold and wants to allow for the initiation of another call from the logical element), the logical element will use one of the available appearance (i.e., idle). If an available appearance is not present, the call will not be accepted by the logical element.

A.1.1 Selected-Standard Appearance

When calls are presented to the logical element, an available (i.e., idle) appearance is selected and only that appearance is presented with the call. From that point on, the handling of the call does not have any special characteristics or behaviours. Figure A-1 illustrates selected-standard appearances. In this example, the logical element is **L** and it has three addressable selected-standard appearances (**A1, A2, A3**). A call is currently being handled by appearance **A3**. When a new call (call 2) is presented to the logical element, it selects one of the available appearances (**A2**) to handle the call.

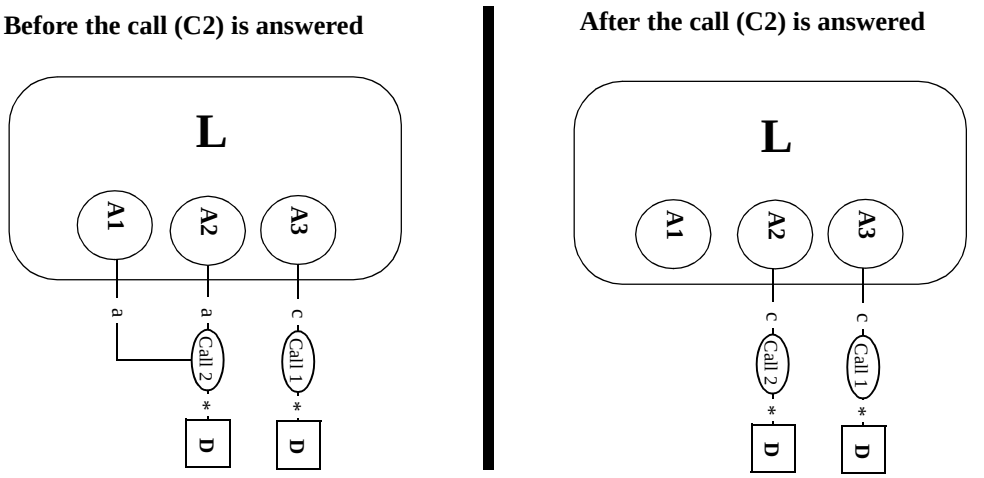
Figure A-1 Selected-Standard Appearances



A.1.2 Basic-Standard Appearance

When calls are presented to the logical element, each available (i.e., idle) appearance is presented with the call. When the call is answered by one of the appearances (i.e., the associated connection state has transitioned to the connected state), the other appearances are cleared from the call. From that point on, the handling of the call does not have any special characteristics or behaviours. The appearances that were cleared from the call can then be used to process other call and call-related activities associated with the logical element. Figure A-2 illustrates basic-standard appearances. In this example, the logical element is **L** and it has three addressable basic-standard appearances (**A1**, **A2**, **A3**). A call is currently being handled by appearance **A3**. When a new call (call 2) is presented to the logical element, it presented to all available appearances (**A2**, **A1**). Appearance **A2** answers the call and appearance **A1** is cleared.

Figure A-2 Basic-Standard Appearances



A.2 Bridged Appearance

A bridged appearance of a device's logical element may be associated with other devices that have a physical element (i.e., can handle calls at another device).

A bridged appearance is a particular implementation of a shared logical element. Note that *associated* in the context of appearances means a relationship between a specific call appearance of a logical element and a device. A call appearance that is associated with a device implies that the call is physically handled at the device's physical element through this call appearance (e.g., makes use of the physical element's auditory apparatus)

These appearances can not be associated with devices that only have a logical element. When a call arrives at the logical element, bridged call appearances are simultaneously selected. Subsequent different behaviours are possible:

- Basic-Bridged (see A.2.1)
- Exclusive-Bridged (see A.2.2)
- Shared-Bridged (see A.2.3)

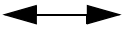
This type of appearance can be either addressable or non-addressable. Refer to Clause 10, “CSTA Device Identifier Formats”, on page 75 for information on how to reference this appearance.

When the appearance is addressable, the association between the appearance and the other device is one to one but the other device may have multiple bridged appearances associated with it. Changes in this association are reflected by the capabilities exchange services (13.1 beginning on page 118). In addition, all media streams associated with calls corresponding to this appearance are only available to the physical element (of the other device that is associated with this appearance).

When the appearance is non-addressable, the association between the appearance and the other device is one to one as long as a call is present at the appearance (i.e., a different device may be associated with this appearance every time a call is present). As a result, the media stream associated with the call is only available to the other device that was assigned to the particular appearance for the call. When a call activity at one appearance results in the need for another appearance (i.e., the appearance puts the call on hold and wants to allow for the initiation of another call from the logical element), the logical element will use one of the available appearance (i.e., idle) at the associated device. If an available appearance at the associated device is not present, the call will not be accepted by the logical element.

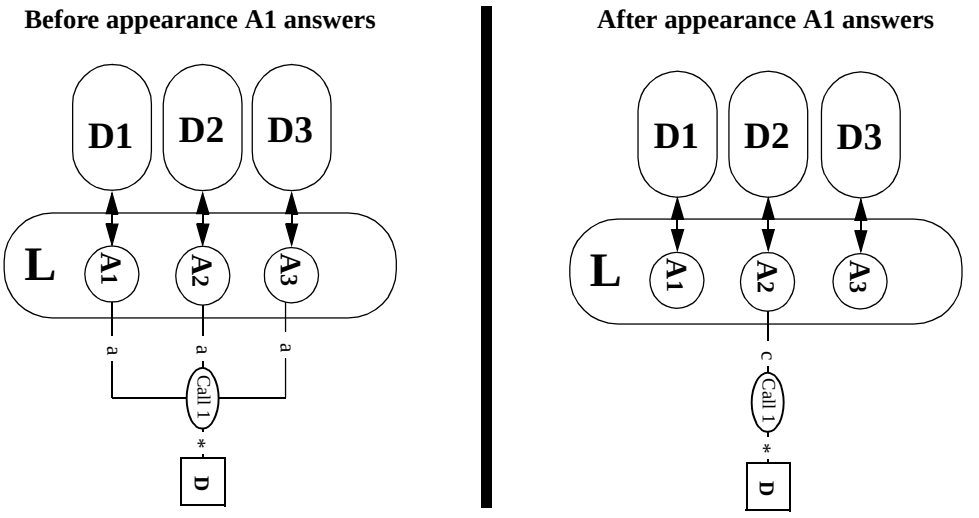
A.2.1 Basic-Bridged

When calls are presented to the logical element, each available (i.e., idle) appearance is presented with the call. When the call is answered by one of the appearances (i.e., the associated connection state has transitioned to the connected state), the other appearances are cleared from the call. From that point on, the handling of the call does not have any special characteristics or behaviours. The appearances that were cleared from the call can then be used to process other call and call-related activities associated with the logical element. Figure A-3 illustrates basic-bridged appearances. In this example, the logical element is **L** and it has three addressable basic-bridged appearances (**A1**, **A2**, **A3**). When a new call (call 1) is presented to the logical element, it is presented to all available appearances (**A1, A2, A3**). Appearance **A2** answers the call and appearances **A3** and **A1** are cleared.



The arrows indicates that the particular bridged appearance of the logical element is associated with the corresponding device.

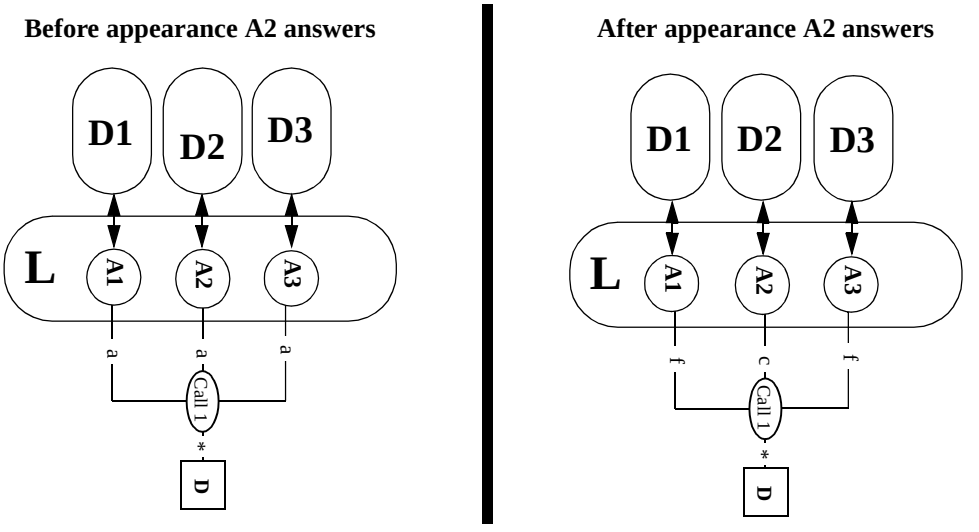
Figure A-3 Basic-Bridged Appearances



A.2.2 Exclusive-Bridged

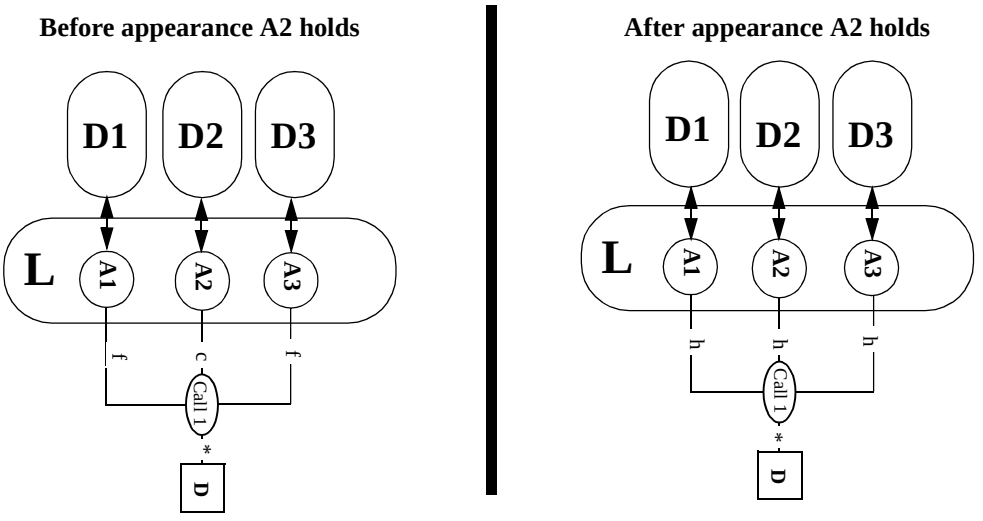
When calls are presented to the logical element, each available (i.e., idle) appearance is presented with the call. When the call is answered by one of the appearances (i.e., the associated connection state has transitioned to the connected state), the other appearances are blocked from being used (i.e., the associated connection state is transitioned to the Failed state) until the connection that answered the call is cleared or the call is moved away. At which time, the connections in the Failed state are cleared. Otherwise, the handling of the call does not have any special characteristics or behaviours. Figure A-4 illustrates exclusive-bridged appearances. In this example, the logical element is **L** has three addressable exclusive-bridged appearances (**A1, A2, A3**). When a new call (call 1) is presented to the logical element, it presented to all available appearances (**A1, A2, A3**). Appearance **A2** answers the call and appearances **A1** and **A3** are blocked.

Figure A-4 Exclusive-Bridged Appearances



When the connected appearance places the call on hold, all other appearances will transition to the hold state. If all appearances are in the hold state and one of appearances retrieves the call, then the appearance retrieving the call will transition to the connected state and the other appearances will transition to the blocked from use mode (i.e., the associated connection state is transitioned to the Failed state). Figure A-5 illustrates this behaviour of exclusive-bridged appearances (Note that this is a continuation of the illustration above). Appearance **A2** places the call on hold while appearances **A2** and **A3** are in the blocked from use mode.

Figure A-5 Exclusive-Bridged Appearances (Continuation of Figure A-4)



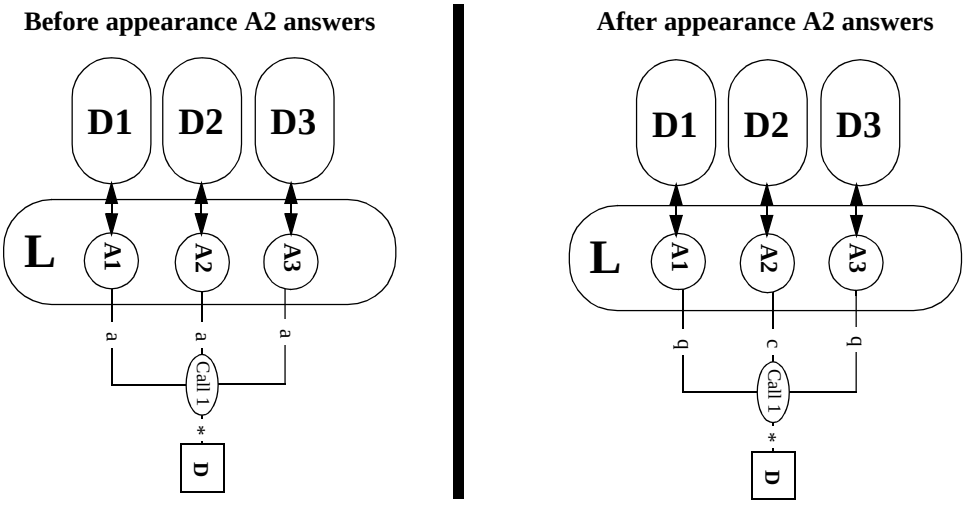
A.2.3 Shared-Bridged

This is the most common form of bridged appearances. When calls are presented to the logical element, each available (i.e., idle) appearance is presented with the call. When the call is answered by one of the appearances (i.e., the associated connection state has transitioned to the connected state), the other appearances enter an inactive mode (i.e., the associated connection state transitions to the queued state) until one of the following actions happen:

- They connect to the call.
- The call is ended.
- They are permanently cleared from the call through some feature/service.
- The call moves away from the device and none of the appearances are connected into the call.

Figure A-6 illustrates this behaviour of shared-bridged appearances. In this example, the logical element is **L** and it has three addressable shared-bridged appearances (**A1, A2, A3**). When a new call (call 1) is presented to the logical element, it presented to all available appearances (**A1, A2, A3**). Appearance **A2** answers the call and appearances **A1** and **A3** are made inactive.

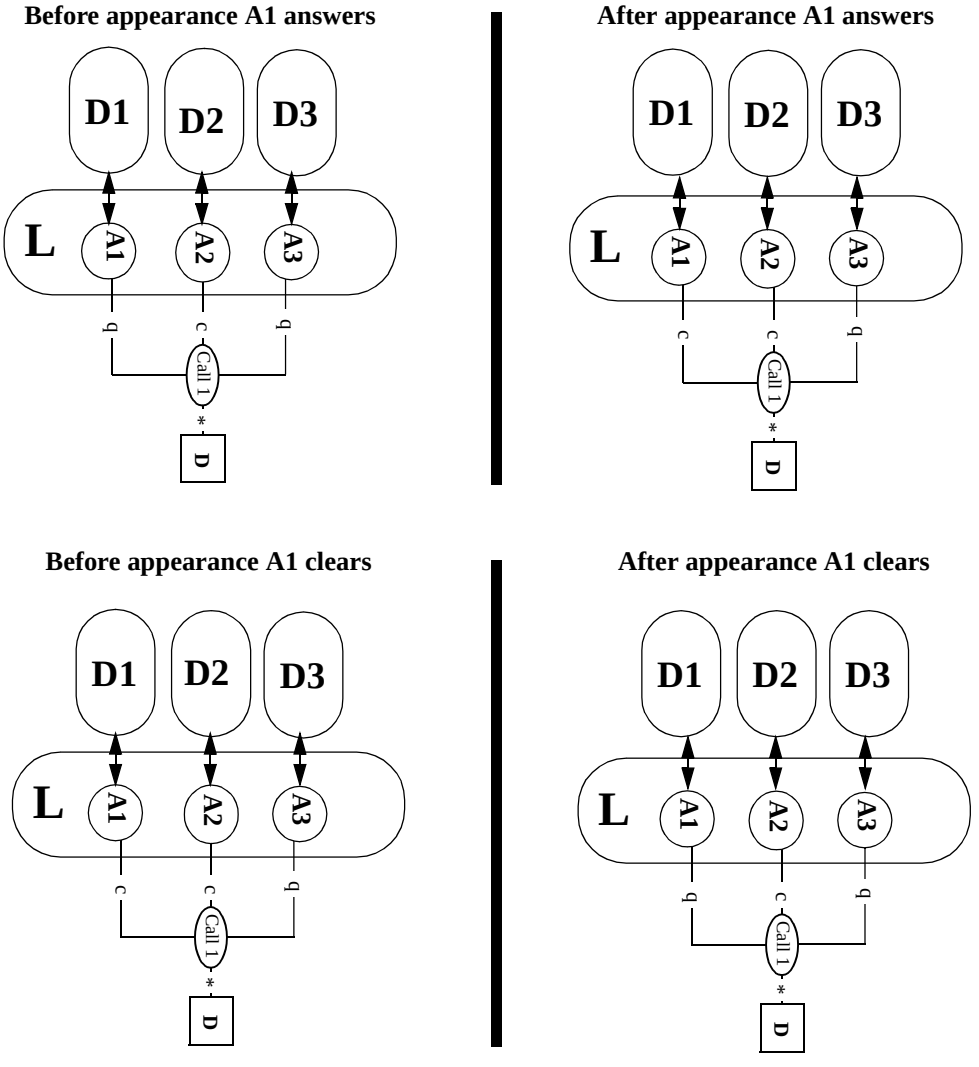
Figure A-6 Shared-Bridged Appearances



The significant behaviour of shared-bridged appearances is that any appearance can join and/or clear from the call at any time up to a switching function limit on the number of appearances connected on the call. As long as at least one of the appearances remains connected to the call, all of the other appearances retain the ability to join the call.

The Answer Call model is used to inform the computing function of appearances joining the call (Established Events). When an appearance clears from the call (with other appearances still connected), the appearance is returned to the inactive mode (i.e., the associated connection state transitions to the queued state). The call ends when all of the appearances clear from the call (i.e., all associated connection states transition to the queued state). Figure A-7 illustrates this behaviour of shared-bridged appearances (Note that this is a continuation of the illustration above). In this case, appearance **A1** adds to and clears from the call while appearance **A2** remains in the call.

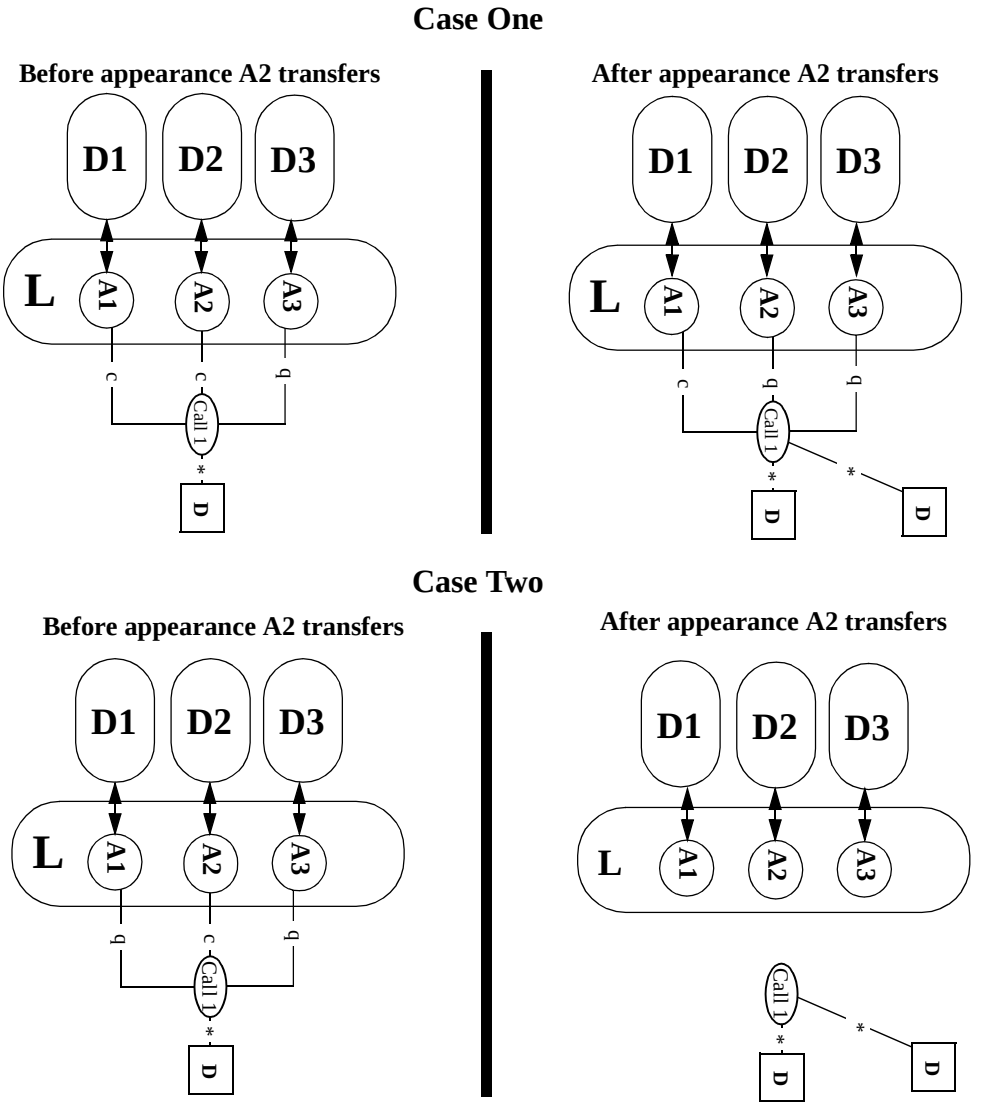
Figure A-7 Shared-Bridged Appearances (Continuation of Figure A-6)



When the call is moved away from an appearance or an appearance places the call on hold, the shared-bridged appearance type becomes two types:

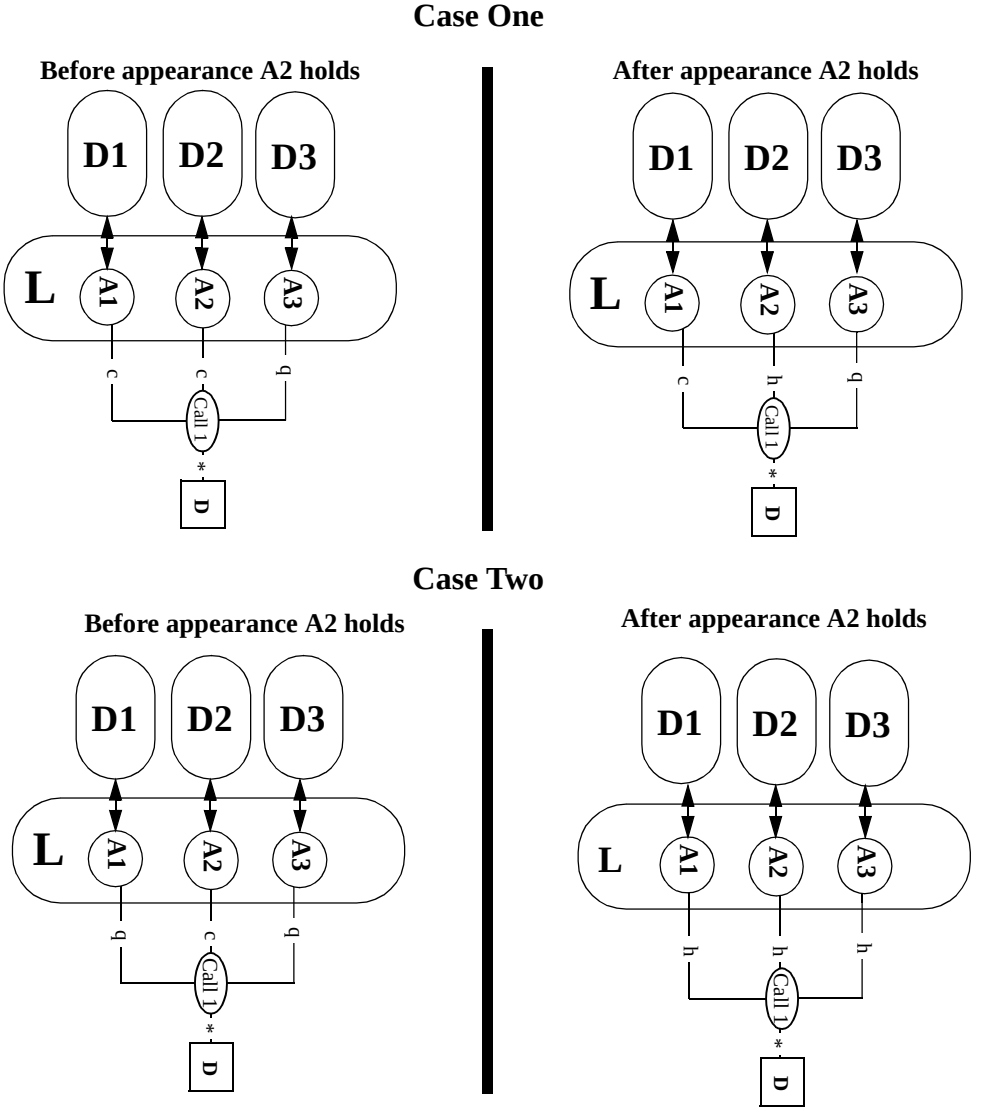
- *Independent-shared-bridged* - When the call is moved away from the logical element by one of the appearances, the other appearances are unaffected, as long as one appearance remains connected in the call, otherwise all appearances are cleared from the call. As for the appearance moving the call, it will return to the inactive mode (if another appearance remains connected in the call). Figure A-8 illustrates this behaviour of independent-shared-bridged appearances (Note that this is a continuation of the illustration above). In case one, appearance **A2** immediately transfers the call to another device while appearance **A1** remains in the call. In case two, appearance **A2** immediately transfers the call to another device while all appearances **A1** and **A3** are in the inactive mode.

Figure A-8 Independent-Shared-Bridged Appearances



When one of the appearances places the call on hold and at least one other appearance is connected in the call, only the holding appearance will transition to the hold state (i.e., the other appearances are unaffected). When one of the appearances places the call on hold and the holding appearance was the only one connected in the call, all other appearances will transition to the hold state. If all appearances are in the hold state and one of appearances retrieves the call, then the appearance retrieving the call will transition to the connected state and the other appearances will transition to the inactive mode. Figure A-9 illustrates this behaviour of independent-shared-bridged appearances (Note that this is a continuation of the illustration above). In case one, appearance **A2** places the call on hold while appearance **A1** remains in the call. In case two, appearance **A2** places the call on hold while all appearances **A2** and **A3** are in the inactive mode.

Figure A-9 Independent-Shared-Bridged Appearances (Continuation of Figure A-8)



- *Interdependent-shared-bridged* - When the call is moved away from the logical element by one of the appearances (no matter what the connection state of the other appearances in the call), all appearances are cleared from the call. Figure A-8 in Case 2 illustrates this behaviour of interdependent-shared-bridged appearances.

  When one of the appearances places the call on hold, all appearances will transition to the hold state. When one of appearances retrieves the call, then the appearance retrieving the call will transition to the connected state and the other appearances will transition to the inactive mode. Figure A-9 in Case 2 illustrates this behaviour of interdependent-shared-bridged appearances.

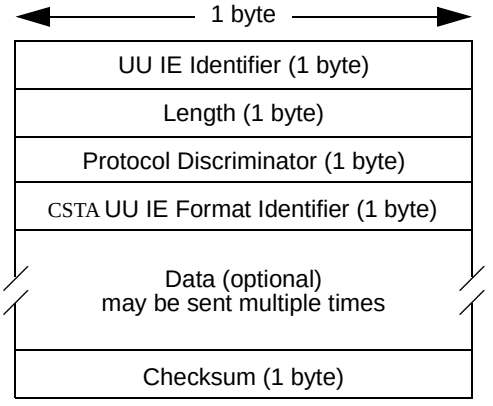
Annex B
(Normative)

ISDN User-User Information Element Encoding for CSTA

Correlator data, user data, or both can be delivered to a switching sub-domain from an ISDN network connection. The data is transmitted through the network in the ISDN User-User (UU) Information Element (IE) in ISDN call control and User Info messages where supported. Transmission of User-User data is not supported by all parts of the public network.

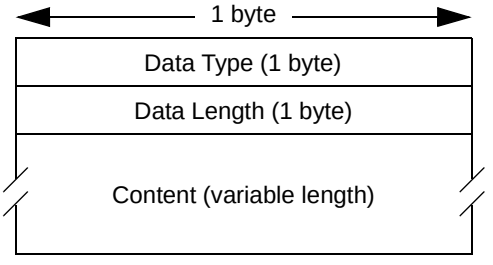
The following CSTA UU IE format shall be used for packing correlator data, user data, or both. The first three bytes of the UU IE are defined in ITU-T Recommendation Q.931.

Figure B-1 CSTA User-User Information Element (UU IE) Field Format



- The *UU IE Identifier* field identifies the UU IE packet. Its value is 0x7E.
- The *Length* field provides the total length of the UU IE packet.
- The *Protocol Discriminator* signals end-to-end transparency to the public switch vendor. Its value is 0x00.
- The CSTA UU IE Format Identifier identifies that the information element contains correlator data, user data, or both. Its value is 0x45.
- The *Data* field is optional, may be sent multiple times and has the following format:

Figure B-2 Data Field Format



- The *Data Type* sub-field specifies whether the data is correlator data (value is 0x80) or User Data (value is 0x81).
- The *Data Length* sub-field provides the length of the Content field.
- The *Content* sub-field contains the appropriate Data (correlator data or user data).
- The *Checksum* field verifies that the packet is the CSTA UU IE packet defined in this section.

Functional Requirements

1. Multiple Data Type entries may be sent in each UU IE. The entries may be repeated up to a maximum packet length of 128 bytes.[1]
2. A switching function reports its supported maximum lengths for Correlator Data and User Data in the capabilities exchange services.
3. The possible maximum data lengths in ISDN messages are 121 bytes for User Data and 32 bytes for Correlator Data.[1]
4. If a switching function does not support a data type, it is discarded. If a switching function does not support as many entries of a particular type as were received, the additional entries are discarded.
5. Length 0 (zero) of Correlator Data is used to signal null Correlator Data to the receiving switching sub-domain. Current Correlator Data associated with the call shall be deleted.
6. Format and interpretation of the Content sub-field is a function of the computing function.
7. The Checksum byte contains the XOR checksum of all the bytes starting with the CSTA UU IE Format Identifier.

1. The maximum size of an ISDN UU IE packet may be limited based upon certain implementations (i.e., some public network implementations).

Annex C
(Normative)

Capability Bitmap Parameters Types

This annex specifies the *capability bitmap parameters types* that are included in the Get Physical Device Information, Get Logical Device Information and Get Switching Function Capabilities services.

These parameter types are summarized in the following table.

Table C-1 Capability Bitmap Parameter Types

Capability Bitmap Parameter Type	Description	Pg.
C.1 CapExchangeServList	Specifies the Capability Exchange services that are supported by the switching function.	578
C.2 SystemStatusServList	Specifies the System Status services that are supported by the switching function.	579
C.3 MonitoringServList	Specifies the Monitoring services that are supported by the switching function.	581
C.4 SnapshotServList	Specifies the Snapshot services that are supported by the switching function.	581
C.5 CallControlServList	Specifies the Call Control services that are supported by the switching function.	583
C.6 CallControlEvtsList	Specifies the Call Control events that are supported by the switching function.	593
C.7 CallAssociatedServList	Specifies the Call Associated services that are supported by the switching function.	599
C.8 CallAssociatedEvtsList	Specifies the Call Control events that are supported by the switching function.	600
C.9 MediaServList	Specifies the Media Attachment services that are supported by the switching function.	601
C.10 MediaEvtsList	Specifies the Media Attachment events that are supported by the switching function.	602
C.11 RouteingServList	Specifies the Routeing services that are supported by the switching function.	603
C.12 PhysDevServList	Specifies the Physical Device Feature services that are supported by the switching function.	605
C.13 PhysDevEvtsList	Specifies the Physical Device Feature events that are supported by the switching function.	609
C.14 LogicalServList	Specifies the Logical Device Feature services that are supported by the switching function.	611
C.15 LogicalEvtsList	Specifies the Logical Device Feature events that are supported by the switching function.	614
C.16 DeviceMaintEvtsList	Specifies the Device Maintenance events that are supported by the switching function.	616
C.17 IOServicesServList	Specifies the I/O services that are supported by the switching function.	617
C.18 DataCollectionServList	Specifies the Data Collection services that are supported by the switching function.	619
C.19 VoiceUnitServList	Specifies the Voice Unit services that are supported by the switching function.	620
C.20 VoiceUnitEvtsList	Specifies the Voice Unit events that are supported by the switching function.	622
C.21 CDRServList	Specifies the CDR services that are supported by the switching function.	624
C.22 VendorSpecificServList	Specifies the Vendor Specific services that are supported by the switching function.	626
C.23 VendorSpecificEvtsList	Specifies the Vendor Specific events that are supported by the switching function.	626

The capability bitmap parameter types specified in this clause consists of sets of entries. A unique entry is defined for each service/event. The switching function provides the appropriate entry for a given service/event if it supports that service/event. It shall not provide an entry for a service/event that is not supported. An entry is a bitmap where each bit represents an option that the switching function either supports or does not support.

The options that are specified in the capability bitmaps include:

- initial connection states supported for a service request. If the operational model for a service specifies more than one possible initial connection state, then the switching function shall indicate which initial connection states it supports for the service request be setting to TRUE the capability bit corresponding to the supported connections states.

- optional parameters. The switching function shall indicate the optional parameters that it supports by setting to TRUE the capability bit corresponding to the supported optional parameter. Note that some optional parameters in services/events are not included in the service/event specific bitmaps since they are specified at a switching function level (via the Get Switching Function Capabilities service). An example of a “global” optional parameter is security type information (timestamp, message sequence numbers, securityInfo).
- conditional parameters. In most cases conditional parameters are not included in the capability bitmaps since, by their definition, the switching function is required to support the parameter under the conditions specified in the parameter description (e.g., support of a specific feature). In some cases where the conditions allow the parameter to be optionally supported then the parameter is included in the capability bitmaps.
- parameter values supported. The switching function shall indicate the set of values that are supported in an enumerated list or in a bitmap list by setting to TRUE the capability bit corresponding to the value supported.
    - for optional parameters that are represented as a bitmap parameter type, there is a bit in the capability list to indicate if the parameter itself is present, and a bit in the capability list for each possible bit in the parameter.
    - for optional parameters that are represented as an enumerated parameter type, there will be a bit in the capability list for each possible value in the parameter (a separate bit to indicate if the parameter itself is present is not necessary).
- miscellaneous characteristics. The switching function shall indicate the miscellaneous characteristics that are supported by setting to TRUE the capability bit corresponding to the characteristic supported.

Note that, in the following lists, items that apply to the service acknowledgement are indicated by the text “in the acknowledgement”, otherwise the item applies to the service request.

C.1 CapExchangeServList

The CapExchangeServList parameter type specifies the capability exchange services supported by the switching function.

The information in the CapExchangeServList represents the optional parameters that are supported by the switching function.

C.1.1 Get Logical Device Information

Optional parameters, values, and choices supported:

- privateData
- namedDeviceTypes in the acknowledgement
- shortFormDeviceID in the acknowledgement
- miscMonitorCaps in the acknowledgement
- maxCallbacks in the acknowledgement
- maxAutoAnswerRings in the acknowledgement
- maxActiveCalls in the acknowledgement
- maxHeldCalls in the acknowledgement
- maxFwdSettings in the acknowledgement
- maxDevicesInConf in the acknowledgement
- transAndConfSetup parameter, transAndConfSetup (consultationCall, holdCallMakeCall, alternateCall, twoCallsHold, twoCallsConnected) in the acknowledgement
- mediaClassSupport in the acknowledgement
- connectionRateList in the acknowledgement
- delayToleranceList in the acknowledgement

- numberofChannels in the acknowledgement
- maxChannelBind in the acknowledgement
- privateData in the acknowledgement

C.1.2 Get Physical Device Information

Optional parameters, values, and choices supported:

- privateData
- namedDeviceTypes in the acknowledgement
- otherLogicalDeviceList in the acknowledgement
- deviceModelName in the acknowledgement
- maxDisplays in the acknowledgement
- maxButtons in the acknowledgement
- maxLamps in the acknowledgement
- maxRingPatterns in the acknowledgement
- privateData in the acknowledgement

C.1.3 Get Switching Function Capabilities

Note that there are no capability bits associated with this service.

C.1.4 Get Switching Function Devices

Optional parameters, values, and choices supported:

- requestedDeviceID
- requestedDeviceCategory (aCD, aCDGroup, huntGroup, pickGroup, otherGroup, networkInterface, park, routeingDevice, station, voiceUnit, other)
- privateData
- privateData in the acknowledgement

C.1.5 Switching Function Devices

Optional parameters, values, and choices supported:

- segmentID
- deviceListDeviceCategory
- deviceListNamedDeviceTypes
- deviceListDeviceAttributes
- deviceListDeviceName
- privateData

C.2 SystemStatusServList

The SystemStatusServList parameter type specifies the system status services supported by the switching function.

C.2.1 Change System Status Filter

Optional parameters, values, and choices supported:

- requestedStatusFilter (initializing, enabled, normal, messagesLost, disabled, partiallyDisabled, overloadImminent, overloadReached, overloadRelieved)
- privateData

- privateData in the acknowledgement

C.2.2 System Register

Optional parameters, values, and choices supported:

- requestTypes (systemStatus, requestSystemStatus)
- requestedStatusFilter (initializing, enabled, normal, messagesLost, disabled, partiallyDisabled, overloadImminent, overloadReached, overloadRelieved)
- privateData
- privateData in the acknowledgement

C.2.3 System Status Register Abort

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.2.4 System Status Register Cancel

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.2.5 Request System Status

Optional parameters, values, and choices supported:

- privateData in the service request
- systemStatus (initializing, enabled, normal, messagesLost, disabled, partiallyDisabled, overloadImminent, overloadReached, overloadRelieved) in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Switching Functions supports sending the service request
- Switching Functions supports receiving the service request

C.2.6 System Status

Optional parameters, values, and choices supported:

- systemStatus (initializing, enabled, normal, messagesLost, disabled, partiallyDisabled, overloadImminent, overloadReached, overloadRelieved) in the service request
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Switching Functions supports sending the service request
- Switching Functions supports receiving the service request

C.2.7 Switching Function Capability Changed

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.2.8 Switching Function Devices Changed

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.3 MonitoringServList

The MonitoringServList parameter type specifies the monitoring services that are supported by the switching function.

C.3.1 Change Monitor Filter

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.3.2 Monitor Start

Optional parameters, values, and choices supported:

- monitorObject (call, device)
- requestedMonitorFilter
- monitorType (callType, deviceType)
- requestedMonitorMediaClass parameter, requestedMonitorMediaClass (audio, data, image, voice)
- monitorExistingCalls in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics

- CallIDOnly - Does the switching function accept or support the CallID only format of the Connection ID for this service?
- Switching function default for monitor-type is device-type (if FALSE then it is call-type).

C.3.3 Monitor Stop

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Does the switching function generate this service request?
- Does the switching function support receiving this service request?

C.4 SnapshotServList

The SnapshotServList parameter type specifies the snapshot services supported by the switching function.

C.4.1 Snapshot Call

Optional parameters, values, and choices supported:

- privateData
- localConnectionState in the acknowledgement
- mediaServiceInformationList in the acknowledgement
    - mediaServiceVersion

    - mediaServiceInstance
    - mediaStreamID
    - connectionInformation
- mediaCallCharacteristics in the acknowledgement
- callCharacteristics in the acknowledgement
- callingDevice in the acknowledgement
- calledDevice in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics

- CallIDOnly - Does the switching function accept or support the CallID only format of the Connection ID for this service?
- Does the switching function use the Snapshot CallData service to report the requested information?

C.4.2 Snapshot Device

Optional parameters, values, and choices supported:

- privateData
- localCallState (compoundCallState, simpleCallState, unknown) in the acknowledgement
- mediaServiceInformationList in the acknowledgement
    - mediaServiceVersion
    - mediaServiceInstance
    - mediaStreamID
    - connectionInformation
- mediaCallCharacteristics in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics

- Does the switching function use the Snapshot DeviceData service to report the requested information?

C.4.3 Snapshot CallData

Optional parameters, values, and choices supported:

- segmentID
- localConnectionState
- mediaServiceInformationList
    - mediaServiceVersion
    - mediaServiceInstance
    - mediaStreamID
    - connectionInformation
- privateData in the acknowledgement

C.4.4 Snapshot DeviceData

Optional parameters, values, and choices supported:

- segmentID

- localCallState (compoundCallState, simpleCallState, unknown)
- mediaServiceInformationList
  - mediaServiceVersion
  - mediaServiceInstance
  - mediaStreamID
  - connectionInformation
- mediaCallCharacteristics
- privateData

C.5 CallControlServList

The CallControlServList parameter specifies the call control services that are supported by the switching function.

C.5.1 Accept Call

Optional parameters, values, and choices supported:

- correlatorData
- userData
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.2 Alternate Call

Initial states heldCall:

- Alerting
- Hold
- Queued

Optional parameters, values, and choices supported:

- connectionReservation
- consultOptions (consultOnly, transferOnly, conferenceOnly, unrestricted)
- privateData
- privateData in the Acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Supports Offered Mode of Alerting?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.3 Answer Call

Initial states callToBeAnswered:

- Alerting

- Initiated
- Queued

Optional parameters, values, and choices supported:

- correlatorData
- userData
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Supports Offered Mode of Alerting?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.4 Call Back Call-Related

Initial states target:

- Alerting
- Fail
- Null
- Queued

Optional parameters, values, and choices supported:

- callCharacteristics
- privateData
- targetDevice in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Do additional Call Back Call-Related services for the same calling and called devices result in a negative acknowledgement? (if FALSE, then it is positively acknowledged).
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.5 Call Back Message Call-Related

Initial states target:

- Alerting
- Fail
- Null
- Queued

Optional parameters, values, and choices supported:

- privateData
- targetDevice in the acknowledgement

- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Do additional Call Back Message Call-Related services for the same calling and called devices result in a negative acknowledgement? (if FALSE, then it is positively acknowledged).
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.6 Camp On Call

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.7 Clear Call

Initial states callToBeCleared:

- Alerting
- Connected
- Fail
- Queued
- Initiated
- Hold

Optional parameters, values, and choices supported:

- userData
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- CallIDOnly - Does the switching function accept or support the CallID only format of the connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.8 Clear Connection

Initial states callToBeCleared:

- Alerting
- Connected
- Fail
- Queued

- Hold
- Initiated

Optional parameters, values, and choices supported:

- correlatorData
- userData
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.9 Conference Call

Initial states activeCall:

- Connected
- Hold

Initial states heldCall:

- Connected
- Hold

Optional parameters, values, and choices supported:

- privateData
- connections parameter in the acknowledgement
  - endpointDeviceID
  - resultingConnectionInformation parameter
- conferenceCallInfo parameter in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Is the resulting conference call protected against being cleared if the conferencing device leaves the call?
- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.10 Consultation Call

Optional parameters, values, and choices supported:

- connectionReservation
- accountCode
- authCode
- correlatorData
- userData

- callCharacteristics, callCharacteristics (acdCall, priorityCall, maintenanceCall, directAgent, assistCall, voiceUnitCall)
- mediaCallCharacteristics
- callingConnectionInfo parameter
  - flowDirection (transmit, receive, transmitAndReceive)
  - numberOfChannels
- consultOptions (consultOnly, transferOnly, conferenceOnly, Unrestricted)
- privateData
- initiatedCallInfo in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- MultiStage- Does the switching function support multistage dialling with this service?
- Does the switching function support adjustment of the media characteristics?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.11 Deflect Call

Initial states callToBeDeflected:

- Alerting
- Connected
- Failed
- Hold
- Queued

Optional parameters, values, and choices supported:

- correlatorData
- userData
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.12 Dial Digits

Optional parameters, values, and choices supported:

- correlatorData
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.13 Directed Pickup Call

Initial states callToBePickedUp:

- Alerting
- Connected
- Hold
- Queued

Optional parameters, values, and choices supported:

- correlatorData
- userData
- privateData
- pickedCall in the acknowledgement
- pickedCallInfo in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Supports Offered Mode of Alerting?
- Prompting - Does the switching function support prompting for the newDestinationDevice?
- Prompting Mode - Is prompting part of the execution of the service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.14 Group Pickup Call

Initial states for the connection that is picked:

- Alerting
- Connected
- Hold
- Queued

Optional parameters, values, and choices supported:

- pickGroup
- correlatorData
- userData
- privateData
- pickedCall in the acknowledgement
- pickedCallInfo in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Prompting - Does the switching function support prompting for the newDestinationDevice?
- Prompting Mode - Is prompting part of the execution of the service?
- Supports Offered Mode of Alerting?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.15 Hold Call

Optional parameters, values, and choices supported:

- connectionReservation
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.16 Intrude Call

Optional parameters, values, and choices supported:

- participationType (active, silent)
- userData
- privateData
- conferencedCallInfo in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Conference - Does the switching function support conferencing the intruder with the existing call?
- Alternate - Does the switching function support putting the existing call on hold and establishing a call with the intruder?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.17 Join Call

Optional parameters, values, and choices supported:

- autoOriginate (prompt, doNotPrompt)
- participationType (active, silent)
- accountCode
- authCode
- correlatorData
- userData
- privateData
- conferencedCall in the acknowledgement
- conferencedCallInfo in the acknowledgement

- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Prompting - Does the switching function support prompting for the joiningDevice?
- Prompting Mode - Is prompting part of the execution of the service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.18 Make Call

Initial states for the calling device connection:

- Initiated
- Null

Optional parameters, values, and choices supported:

- accountCode
- authCode
- autoOriginate (prompt, doNotPrompt)
- correlatorData
- userData
- callCharacteristics parameter, callCharacteristics (acdCall, priorityCall, maintenanceCall, directAgent, assistCall, voiceUnitCall)
- mediaCallCharacteristics
- callingConnectionInfo
- privateData
- initiatedCallInfo in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- MultiStage- Does the switching function support multistage dialling with this service?
- Prompting - Does the switching function support prompting for the callingDevice?
- Prompting Mode - Is prompting part of the execution of the service?
- Offhook - Does the switching function support the performing of a Make Call while the callingDevice is off-hook?
- Can the switching function adjust the media characteristics of the call?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.19 Make Predictive Call

Optional parameters, values, and choices supported:

- signallingDetection parameter
  - signallingCondition (callDelivered, CallEstablished)
  - signallingConditionsAction (destinationDetection, remainConnected)

- destinationDetection parameter
  - destinationCondition (humanVoice, answeringMachine, facimileMachine)
  - detectionAction (clearCallConnection, remainConnected)
- defaultAction (clearCalledConnection, remainConnected)
- accountCode
- authCode
- autoOriginate (prompt, doNotPrompt)
- alertTime
- correlatorData
- callCharacteristics, callCharacteristics (acdCall, priorityCall, maintenanceCall, directAgent, assistCall, voiceUnitCall)
- userData
- privateData
- initiatedCallInfo in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Prompting - Does the switching function support prompting for the callingDevice?
- Prompting Mode - Is prompting part of the execution of the service?
- Does the switching function generate a Service Initiated event for the calling device prior to any call activity at the calling device (i.e. reserve the calling device)?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.20 Park Call

Initial states callToBeParked:

- Connected
- Hold

Optional parameters, values, and choices supported:

- correlatorData
- privateData
- parkedTo in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.21 Reconnect Call

Initial states activeCall:

- Alerting
- Connected

- Fail
- Initiated
- Queued

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.22 Retrieve Call

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.23 Single Step Conference Call

Optional parameters, values, and choices supported:

- participationType (active, silent)
- accountCode
- authCode
- correlatorData
- userData
- privateData
- conferencedCallInfo in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.24 Single Step Transfer Call

Optional parameters, values, and choices supported:

- accountCode
- authCode
- correlatorData
- userData

- privateData
- connections parameter in the acknowledgement
  - endpointDeviceID
  - resultingConnectionInformation
- transferredCallInfo in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- MultipleDevices - Does the switching function support transferring multiple devices or a conference call?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.5.25 Transfer Call

Initial states activeCall:

- Connected
- Hold

Initial states heldCall:

- Connected
- Hold

Optional parameters, values, and choices supported:

- privateData
- connections parameter in the acknowledgement
  - endpointDeviceID
  - resultingConnectionInformation
- transferredCallInfo in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- MultipleDevices - Does the switching function support transferring multiple devices or a conference call?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.6 CallControlEvtsList

The CallControlEvtsList parameter type specifies the call control events that are supported by the switching function.

The following list describes some general requirements for the capability options related to the Call Control events.

1. The localConnectionInfo parameter is not specified in the capability bit maps for the events. The computing function can use the Get Switching Function Capabilities service to determine if device-type monitoring is supported.
2. The conditional parameter correlatorData is not specified in the capability bit maps for the events since it is required if the switching function supports correlator data. The computing function can use the Get Switching

Function Capabilities service to determine if the correlator data feature is supported. Note that for the events where the correlatorData has been defined as optional, the capability bit is specified.

C.6.1 Bridged

Optional parameters, values, and choices supported:

- correlatorData
- userData
- servicesPermitted
- mediaCallCharacteristics
- callCharacteristics
- bridgedConnectionInfo
- privateData

C.6.2 Call Cleared

Optional parameters, values, and choices supported:

- correlatorData
- userData
- mediaCallCharacteristics
- callCharacteristics
- privateData

Miscellaneous Characteristics:

- CallIDOnly - Does the switching function accept or support the CallID only format of the Connection ID for this event?

C.6.3 Conferenced

Optional parameters, values, and choices supported:

- conferenceConnections (endpointDeviceID, resultingConnectionInformation)
- userData
- servicesPermitted
- mediaCallCharacteristics
- callCharacteristics
- privateData

C.6.4 Connection Cleared

Optional parameters, values, and choices supported:

- correlatorData
- userData
- chargingInfo
  - numberUnits (numberOfChargingUnits, typeOfUnits, numberOfCurrencyUnits)
  - typeOfChargingInformation (subTotal, total)
  - chargingMultiplier (.001, .01, .1, 1, 10, 100, 1000)
- servicesPermitted

- mediaCallCharacteristics
- callCharacteristics
- droppedConnectionInfo
- privateData

C.6.5 Delivered

Optional parameters, values, and choices supported:

- originatingNIDConnection
- userData
- servicesPermitted
- networkCallingDevice
- networkCalledDevice
- mediaCallCharacteristics
- callCharacteristics
- connectionInfo
- privateData

C.6.6 Digits Dialed

Optional parameters, values, and choices supported:

- servicesPermitted
- networkCallingDevice
- networkCalledDevice
- diallingConnectionInfo
- callCharacteristics
- privateData

C.6.7 Diverted

Optional parameters, values, and choices supported:

- callingDevice
- calledDevice
- userData
- servicesPermitted
- mediaCallCharacteristics
- callCharacteristics
- connectionInfo
- networkCallingDevice
- networkCalledDevice
- privateData
- Does the switching function send the Diverted Event to all devices in a call (for Device-type monitor) and to a Call-type monitor?

C.6.8 Established

Optional parameters, values, and choices supported:

- originatingNIDConnection
- userData
- servicesPermitted
- networkCallingDevice
- networkCalledDevice
- mediaCallCharacteristics
- callCharacteristics
- establishedConnectionInfo
- privateData

C.6.9 Failed

Optional parameters, values, and choices supported:

- originatingNIDConnection
- userData
- servicesPermitted
- networkCallingDevice
- networkCalledDevice
- mediaCallCharacteristics
- callCharacteristics
- failedConnectionInfo
- privateData

Miscellaneous Characteristics:

- CallIDOnly - Does the switching function support the CallID only format of the Connection ID for this event?

C.6.10 Held

Optional parameters, values, and choices supported:

- correlatorData
- servicesPermitted
- mediaCallCharacteristics
- callCharacteristics
- heldConnectionInfo
- privateData

C.6.11 Network Capabilities Changed

Optional parameters, values, and choices supported:

- progressLocation (user, privateServingLocal, publicServingLocal, transitNetwork, publicServingRemote, privateServingRemote, local, international, networkBeyondInterworkingPoint, other)
- progressDescription (iSDN, qSIG, other)
- userData

- networkType (iSDNPublic, nonISDNPublic, iSDNPrivate, nonISDNPrivate, other)
- eventsProvided parameter, eventsProvided (bridged, callCleared, conferenced, connectionCleared, delivered, digitsDialled, diverted, established, failed, held, networkCapabilitiesChanged, networkReached, offered, originated, queued, retrieved, serviceInitiated, transferred)
- servicesPermitted
- mediaCallCharacteristics
- callCharacteristics
- outboundConnectionInfo
- privateData

C.6.12 Network Reached

Optional parameters, values, and choices supported:

- originatingNIDConnection
- userData
- networkType (iSDNPublic, nonISDNPublic, iSDNPrivate, nonISDNPrivate, other)
- eventsProvided parameter, eventsProvided (bridged, callCleared, conferenced, connectionCleared, delivered, digitsDialled, diverted, established, failed, held, networkCapabilitiesChanged, networkReached, offered, originated, queued, retrieved, serviceInitiated, transferred)
- servicesPermitted
- mediaCallCharacteristics
- callCharacteristics
- outboundConnectionInfo
- networkCallingDevice
- networkCalledDevice
- privateData

C.6.13 Offered

Optional parameters, values, and choices supported:

- originatingNIDConnection
- userData
- servicesPermitted
- networkCallingDevice
- networkCalledDevice
- mediaCallCharacteristics
- callCharacteristics
- offeredConnectionInfo
- privateData

C.6.14 Originated

Optional parameters, values, and choices supported:

- originatingDevice
- servicesPermitted

- networkCallingDevice
- networkCalledDevice
- mediaCallCharacteristics
- callCharacteristics
- originatedConnectionInfo
- privateData

C.6.15 Queued

Optional parameters, values, and choices supported:

- numberQueued
- callsInFront
- userData
- servicesPermitted
- networkCallingDevice
- networkCalledDevice
- mediaCallCharacteristics
- callCharacteristics
- queuedConnectionInfo
- privateData

C.6.16 Retrieved

Optional parameters, values, and choices supported:

- correlatorData
- servicesPermitted
- mediaCallCharacteristics
- callCharacteristics
- retrievedConnectionInfo
- privateData

C.6.17 Service Initiated

Optional parameters, values, and choices supported:

- servicesPermitted
- mediaCallCharacteristics
- callCharacteristics
- initiatedConnectionInfo
- networkCallingDevice
- networkCalledDevice
- privateData

C.6.18 Transferred

Optional parameters, values, and choices supported:

- transferredConnections (endpointDeviceID, resultingConnectionInformation)

- userData
- chargingInfo
  - numberUnits (numberOfChargingUnits, typeOfUnits, numberOfCurrencyUnits)
  - typeOfChargingInformation (subTotal, total)
  - chargingMultiplier (.001, .01, .1, 1, 10, 100, 1000)
- servicesPermitted
- mediaCallCharacteristics
- callCharacteristics
- connectionInfo
- privateData

C.7 CallAssociatedServList

The CallAssociatedServList parameter type specifies the call associated services that are supported by the switching function.

C.7.1 Associate Data

Optional parameters, values, and choices supported:

- accountCode
- authCode
- correlatorData
- callQualifyingData
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Does the switching function reject Associate Data service requests (for accountCode or callQualifyingData) that contain a connectionID that no longer exists?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.7.2 Cancel Telephony Tone

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.7.3 Generate Digits

Optional parameters, values, and choices supported:

- digitMode (rotaryPulse, dTMF)
- toneDuration
- pulseRate
- pauseDuration

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Does the switching function support the DTMF tones “A, B, C, D”?
- Does the switching function support the pause tone character of “;”?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.7.4 Generate Telephony Tones

Optional parameters, values, and choices supported:

- toneToSend (beep, billing, busy, carrier, confirmation, dial, faxCNG, hold, howler, intrusion, modemCNG, park, recordWarning, reorder, ringback, silence, sitVC, sitIC, sitRO, sitNC, sf0 through sf100)
- toneDuration
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.7.5 Send User Information

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.8 CallAssociatedEvtsList

The CallAssociatedEvtsList parameter type specifies the call associated events that are supported by the switching function.

C.8.1 Call Information

Optional parameters, values, and choices supported:

- callingDevice
- accountInfo
- authorisationCode
- correlatorData
- servicesPermitted
- userData
- callQualifyingData

- connectionInfo
- privateData

Miscellaneous Characteristics:

- Does the switching function generate Call Information events (for accountCode or callQualifyingData) that contain a connectionID that no longer exists?

C.8.2 Charging

Optional parameters, values, and choices supported:

- numberUnits (numberOfChargingUnits, typeOfUnits, numberOfCurrencyUnits)
- typeOfChargingInformation (subTotal, total)
- chargingMultiplier (.001, .01, .1, 1, 10, 100, 1000)
- privateData

C.8.3 Digits Generated

Optional parameters, values, and choices supported:

- digitDurationList
- pauseDurationList
- connectionInfo
- privateData

C.8.4 Telephony Tones Generated

Optional parameters, values, and choices supported:

- toneGenerated (beep, billing, busy, carrier, confirmation, dial, faxCNG, hold, howler, intrusion, modemCNG, park, recordWarning, reorder, ringback, silence, sitVC, sitIC, sitRO, sitNC, sf0 through sf100, other, unknown)
- toneFrequency
- toneDuration
- pauseDuration
- connectionInfo
- privateData

C.8.5 Service Completion Failure

Optional parameters, values, and choices supported:

- primaryCallConnectionInfo
- secondaryCall parameter, secondaryCallConnectionInfo
- otherDevicesPrimaryCallList parameter, otherDevicesPrimaryCallListConnectionInfo
- otherDevicesSecondaryCallList parameter, otherDevicesSecondaryCallListConnectionInfo
- mediaCallCharacteristics
- privateData

C.9 MediaServList

The MediaServList parameter type specifies the media attachment services that are supported by the switching function.

C.9.1 Attach Media Service

Optional parameters, values, and choices supported:

- mediaServiceVersion
- mediaServiceInstanceID
- connectionMode (consultationConference, consultationConferenceHold, deflect, directedPickup, join, singleStepConference, singleStepConferenceHold, singleStepTransfer, transfer, direct)
- requestedConnectionState
- privateData
- mediaServiceInstanceID in the acknowledgement
- mediaConnectionInfo in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.9.2 Detach Media Service

Initial States connection

- Alerting
- Connected
- Fail
- Hold
- Queued

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- DeviceIDOnly - Does the switching function accept or support the DeviceID only format of the Connection ID for this service?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.10 MediaEvtsList

The MediaEvtsList parameter type specifies the media attachment services that are supported by the switching function.

C.10.1 Media Attached

Optional parameters, values, and choices supported:

- mediaServiceVersion
- mediaServiceInstanceID
- mediaStreamID
- mediaCallCharacteristics
- callCharacteristics

- mediaConnectionInfo
- privateData

C.10.2 Media Detached

Optional parameters, values, and choices supported:

- mediaServiceVersion
- mediaServiceInstanceID
- mediaStreamID
- mediaCallCharacteristics
- callCharacteristics
- mediaConnectionInfo
- privateData

C.11 RouteingServList

The RouteingServList parameter type specifies the routeing services which the switching function might request from the computing function or that the switching function supports from the computing function.

C.11.1 Route Register

Optional parameters, values, and choices supported:

- routeingDevice
- requestedMonitorMediaClass parameter, requestedMonitorMediaClass (audio, data, image, voice)
- privateData
- actualRouteingMediaClass in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- AllrouteingDevices - Does the switching function support the ability to register for all routeing devices within the switching function?

C.11.2 Route Register Abort

Optional parameters, values, and choices supported:

- privateData

C.11.3 Route Register Cancel

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.11.4 Re-Route

Optional parameters, values, and choices supported:

- replyTimeout
- correlatorData
- privateData

C.11.5 Route End

Optional parameters, values, and choices supported:

- errorValue
- correlatorData
- privateData

Miscellaneous Characteristics:

- Switching function supports sending this service request to the computing function?
- Switching function supports receiving this service request from the computing function?

C.11.6 Route Reject

Optional parameters, values, and choices supported:

- rejectCause (busyOverflow, queueTimeOverflow, capacityOverflow, calanderOverflow, unknownOverflow)
- correlatorData
- privateData

C.11.7 Route Request

Optional parameters, values, and choices supported:

- callingDevice
- calledDevice
- routeSelAlgorithm (aCD, emergency, leastCost, normal, userDefined)
- priority
- replyTimeout
- correlatorData
- mediaCallCharacteristics
- callCharacteristics
- routedCallInfo
- privateData

Miscellaneous Characteristics:

- Does the switching function use the Route Request for non-call related routeing?

C.11.8 Route Select

Optional parameters, values, and choices supported:

- remainRetries (noListAvailable, noCountAvailable, retryCount)
- routeUsedReq
- correlatorData
- privateData

C.11.9 Route Used

Optional parameters, values, and choices supported:

- callingDevice
- domain
- correlatorData
- privateData

C.12 PhysDevServList

The PhysDevServList parameter type specifies the physical device feature services that are supported by the switching function.

C.12.1 Press Button

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.12.2 Get Auditory Apparatus Information

Optional parameters, values, and choices supported:

- auditoryApparatus
- privateData
- auditoryApparatusType (speakerphone, handset, headset, speakerOnlyPhone, other)
- speaker (present, volumeSettable, volumeReadable, muteSettable, muteReadable)
- microphone (present, gainSettable, gainReadable, muteSettable, muteReadable)
- hookswitch (hookswitchSettable, hookswitchOnhook)
- privateData in the acknowledgement

C.12.3 Get Button Information

Optional parameters, values, and choices supported:

- button
- privateData
- buttonLabel in the acknowledgement
- buttonLabelSettable in the acknowledgement
- buttonFunction in the acknowledgement
- buttonAssociatedNumber in the acknowledgement
- buttonAssociatedNumberSettable in the acknowledgement
- buttonPressIndicator in the acknowledgement
- lampList in the acknowledgement
- privateData in the acknowledgement

C.12.4 Get Display

Optional parameters, values, and choices supported:

- displayID
- privateData
- characterSet (aSCII, unicode, proprietary) in the acknowledgement
- privateData in the acknowledgement

C.12.5 Get HookSwitch Status

Optional parameters, values, and choices supported:

- hookswitch
- privateData
- privateData in the acknowledgement

C.12.6 Get Lamp Information

Optional parameters, values, and choices supported:

- lamp
- privateData
- lampLabel in the acknowledgement
- button in the acknowledgement
- lampColor in the acknowledgement
- privateData in the acknowledgement

C.12.7 Get Lamp Mode

Optional parameters, values, and choices supported:

- lamp
- privateData
- lampMode (brokenFlutter, flutter, off, steady, wink) in the acknowledgement
- lampBrightness (unspecifiedNormal, dim, bright) in the acknowledgement
- lampColor in the acknowledgement
- button in the acknowledgement
- privateData in the acknowledgement

C.12.8 Get Message Waiting Indicator

Optional parameters, values, and choices supported:

- privateData
- deviceForMessage in the acknowledgement
- lampIsPresent in the acknowledgement
- privateData in the acknowledgement

C.12.9 Get Microphone Gain

Optional parameters, values, and choices supported:

- auditoryApparatus
- privateData
- micGainAbs in the acknowledgement
- privateData in the acknowledgement

C.12.10 Get Microphone Mute

Optional parameters, values, and choices supported:

- auditoryApparatus
- privateData
- privateData in the acknowledgement

C.12.11 Get Ringer Status

Optional parameters, values, and choices supported:

- ringer
- privateData
- ringCount in the acknowledgement
- ringPattern in the acknowledgement
- ringVolume in the acknowledgement
- ringVolAbs in the acknowledgement
- privateData in the acknowledgement

C.12.12 Get Speaker Mute

Optional parameters, values, and choices supported:

- auditoryApparatus
- privateData
- privateData in the acknowledgement

C.12.13 Get Speaker Volume

Optional parameters, values, and choices supported:

- auditoryApparatus
- privateData
- speakerVolAbs in the acknowledgement
- privateData in the acknowledgement

C.12.14 Set Button Information

Optional parameters, values, and choices supported:

- buttonLabel
- buttonAssociatedNumber
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.12.15 Set Display

Optional parameters, values, and choices supported:

- physicalBaseRowNumber
- physicalBaseColumnNumber
- offset
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Does the switching function support modifying the relative positions of the logical and physical displays (i.e., scrolling) via this service?

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.12.16 Set HookSwitch Status

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.12.17 Set Lamp Mode

Optional parameters, values, and choices supported:

- lampMode (brokenFlutter, flutter, off, steady, wink)
- lampBrightness (unspecifiedNormal, dim, bright)
- lampColor (noColor, red, yellow, green, blue, reserved, sf0 through sf100)
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.12.18 Set Message Waiting Indicator

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.12.19 Set Microphone Gain

Optional parameters, values, and choices supported:

- microphoneGain (micGainAbs, micGainInc)
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.12.20 Set Microphone Mute

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.12.21 Set Ringer Status

Optional parameters, values, and choices supported:

- ringerMode (ringing, not ringing)
- ringVolume (ringVolAbs, ringVolInc)
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.12.22 Set Speaker Mute

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.12.23 Set Speaker Volume

Optional parameters, values, and choices supported:

- speakerVolume (speakerVolAbs, speakerVolInc)
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])
- The speaker volume is reset after every call.
- The setting of the speaker volume is not allowed while participating in an active call.

C.13 PhysDevEvtsList

The PhysDevEvtsList parameter type specifies the physical device feature events that are supported by the switching function.

C.13.1 Button Information

Optional parameters, values, and choices supported:

- buttonLabel
- buttonAssociatedNumber
- buttonPressIndicator
- privateData

C.13.2 Button Press

Optional parameters, values, and choices supported:

- buttonLabel
- buttonAssociatedNumber
- privateData

C.13.3 Display Updated

Optional parameters, values, and choices supported:

- characterSet (aSCII, unicode, proprietary)

- privateData

C.13.4 Hookswitch

Optional parameters, values, and choices supported:

- privateData

C.13.5 Lamp Mode

Optional parameters, values, and choices supported:

- lampMode (brokenFlutter, flutter, off, steady, wink)
- lampBrightness (unspecifiedNormal, dim, bright)
- lampColor (noColor, red, yellow, green, blue, unknown, sf0 through sf100)
- privateData

C.13.6 Message Waiting

Optional parameters, values, and choices supported:

- deviceForMessage
- privateData

C.13.7 Microphone Gain

Optional parameters, values, and choices supported:

- microphoneGain (micGainAbs, micGainInc)
- privateData

C.13.8 Microphone Mute

Optional parameters, values, and choices supported:

- privateData

C.13.9 Ringer Status

Optional parameters, values, and choices supported:

- ringerMode (ringing, not ringing)
- ringCount
- ringPattern
- ringVolume (ringVolAbs, ringVolInc)
- privateData

C.13.10 Speaker Mute

Optional parameters, values, and choices supported:

- privateData

C.13.11 Speaker Volume

Optional parameters, values, and choices supported:

- speakerVolume (speakerVolAbs, speakerVolInc)
- privateData

C.14 LogicalServList

The LogicalServList parameter type specifies the logical device feature services that are supported by the switching function.

C.14.1 Call Back Non-Call-Related

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Do additional Call Back Non-Call-Related service requests for the same originating and target devices result in a negative acknowledgement?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.14.2 Call Back Message Non-Call-Related

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Do additional Call Back Message Non-Call-Related service requests for the same originating and target devices result in a negative acknowledgement?
- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.14.3 Cancel Call Back

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])
- Does the switching function support clearing of all Call Back features at a device (e.g., supporting a null format Device ID)?

C.14.4 Cancel Call Back Message

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])
- Does the switching function support clearing of all Call Back Message features at a device (e.g., supporting a null format Device ID)?

C.14.5 Get Agent Status

Optional parameters, values, and choices supported:

- acdGroup

- privateData
- agentStateList (Agent ID) in the acknowledgement
- agentInfo (pendingAgentState, agentStateCondition (forcedPause, pause)) in the acknowledgement
- privateData in the acknowledgement

C.14.6 Get Auto Answer

Optional parameters, values, and choices supported:

- privateData
- numberOfRings in the acknowledgement
- privateData in the acknowledgement

C.14.7 Get Auto Work Mode

Optional parameters, values, and choices supported:

- privateData
- autoWorkInterval in the acknowledgement
- privateData in the acknowledgement

C.14.8 Get Caller ID Status

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.14.9 Get Do Not Disturb

Optional parameters, values, and choices supported:

- privateData
- callOrigination parameter, callOrigination (internal, external) in the acknowledgement
- callingDeviceList in the acknowledgement
- privateData in the acknowledgement

C.14.10 Get Forwarding

Optional parameters, values, and choices supported:

- privateData
- forwardList (forwardImmediate, forwardBusy, forwardDND, forwardNoAns, forwardBusyInt, forwardBusyExt, forwardDNDInt, forwardDNDExt, forwardNoAnsInt, forwardNoAnsExt, forwardImmInt, forwardImmExt) in the acknowledgement
- forwardDN in the acknowledgement
- forwardDefault (defaultForwardingTypeAndForwardingDN, defaultForwardingType, defaultForwardDN) in the acknowledgement
- ringCount in the acknowledgement
- privateData in the acknowledgement

C.14.11 Get Last Number Dialed

Optional parameters, values, and choices supported:

- privateData

- privateData in the acknowledgement

C.14.12 Get Routeing Mode

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.14.13 Set Agent State

Optional parameters, values, and choices supported:

- requestedAgentState (loggedOn, loggedOff, notReady, ready, workingAfterCall)
- agentID
- password
- group
- privateData
- pendingAgentState (workingAfterCall, notReady, null) in the acknowledgement
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])
- Does the switching function allow a group (ACD group) device in the service request (i.e. service applies to all agents associated with the ACD group)?
- Does the switching function allow an ACD device in the service request (i.e. service applies to all agents associated with the ACD device)?
- Does the switching function delay transition to the requestedAgentState if it is Busy (i.e. support the pending agent state)?
- Does the switching function delay transition to the requestedAgentState if it is WorkingAfterCall(i.e. support the pending agent state)?

C.14.14 Set Auto Answer

Optional parameters, values, and choices supported:

- numberOfRings
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.14.15 Set Auto Work Mode

Optional parameters, values, and choices supported:

- autoWorkInterval
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

- Does the switching function allow a group (ACD group) device in the service request (i.e. service applies to all agents associated with the ACD group)?
- Does the switching function allow an ACD device in the service request (i.e. service applies to all agents associated with the ACD device)?

C.14.16 Set Caller ID Status

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.14.17 Set Do Not Disturb

Optional parameters, values, and choices supported:

- callOrigination parameter, callOrigination (internal, external)
- callingDeviceList
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.14.18 Set Forwarding

Optional parameters, values, and choices supported:

- forwardingType (forwardImmediate, forwardBusy, forwardDND, forwardNoAns, forwardBusyInt, forwardBusyExt, forwardDNDInt, forwardDNDExt, forwardNoAnsInt, forwardNoAnsExt, forwardImmInt, forwardImmExt)
- forwardDN
- ringCount
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.14.19 Set Routeing Mode

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- Service Request Acknowledgement Model (Atomic [FALSE] or Multi-Step [TRUE])

C.15 LogicalEvtsList

The LogicalEvtsList parameter type specifies the logical device feature services that are supported by the switching function.

C.15.1 Agent Busy

Optional parameters, values, and choices supported:

- agentID
- acdGroup
- pendingAgentState (workingAfterCall, notReady, ready, null)
- cause
- privateData

C.15.2 Agent Logged Off

Optional parameters, values, and choices supported:

- agentID
- acdGroup
- agentPassword
- cause
- privateData

C.15.3 Agent Logged On

Optional parameters, values, and choices supported:

- agentID
- acdGroup
- agentPassword
- cause
- privateData

C.15.4 Agent Not Ready

Optional parameters, values, and choices supported:

- agentID
- acdGroup
- cause
- privateData

C.15.5 Agent Ready

Optional parameters, values, and choices supported:

- agentID
- acdGroup
- cause
- privateData

C.15.6 Agent Working After Call

Optional parameters, values, and choices supported:

- agentID
- acdGroup

- pendingAgentState (notReady, ready, null)
- cause
- privateData

C.15.7 Auto Answer

Optional parameters, values, and choices supported:

- numberOfRings
- privateData

C.15.8 Auto Work Mode

Optional parameters, values, and choices supported:

- privateData

C.15.9 Call Back

Optional parameters, values, and choices supported:

- privateData

C.15.10 Call Back Message

Optional parameters, values, and choices supported:

- privateData

C.15.11 Caller ID Status

Optional parameters, values, and choices supported:

- privateData

C.15.12 Do Not Disturb

Optional parameters, values, and choices supported:

- callOrigination parameter, callOrigination (internal, external)
- callingDeviceList
- privateData

C.15.13 Forwarding

Optional parameters, values, and choices supported:

- forwardingType (forwardImmediate, forwardBusy, forwardNoAns, forwardDND, forwardBusyInt, forwardNoAnsInt, forwardNoAnsExt, forwardImmInt, forwardImmExt, forwardDNDInt, forwardDNDExt)
- forwardTo
- forwardDefault (defaultForwardingTypeAndForwardDN, defaultForwardingType, defaultForwardDN)
- ringCount
- privateData

C.15.14 Routeing Mode

Optional parameters, values, and choices supported:

- privateData

C.16 DeviceMaintEvtsList

The DevicemaintenanceEvtsList parameter type specifies the device maintenance events that are supported by the switching function.

C.16.1 Back In Service

Optional parameters, values, and choices supported:

- cause
- privateData

C.16.2 Device Capabilities Changed

Optional Parameters

- cause
- privateData

C.16.3 Out of Service

Optional parameters, values, and choices supported:

- cause
- privateData

C.17 IOServicesServList

The IOServicesServList parameter type specifies the I/O services that are supported by the switching function.

C.17.1 I/O Register

- ioDevice
- privateData
- privateData in the acknowledgement

Miscellaneous Characteristics:

- allIODevices - Does the switching function support the ability to register for all I/O devices within the switching function?

C.17.2 I/O Register Abort

Optional parameters, values, and choices supported:

- privateData

C.17.3 I/O Register Cancel

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.17.4 Data Path Resumed

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.17.5 Data Path Suspended

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.17.6 Fast Data

Optional parameters, values, and choices supported:

- object (device, call)
- dataPathType (text, voice)
- displayAttributes (physicalBaseRowNumber, physicalBaseColumnNumber, offset)
- privateData
- privateData in the acknowledgement

Miscellaneous characteristics:

- Does the switching function support modifying the relative positions of the logical and physical displays (i.e., scrolling) via this service?

C.17.7 Resume Data Path

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

Miscellaneous characteristics:

- Does the switching function send the Data Path Resumed service as the result of this service request?

C.17.8 Send Broadcast Data

Optional parameters, values, and choices supported:

- privateData
- dataPathType (text, voice)
- displayAttributes (physicalBaseRowNumber, physicalBaseColumnNumber, offset)
- privateData in the acknowledgement

Miscellaneous characteristics:

- Does the switching function support modifying the relative positions of the logical and physical displays (i.e., scrolling) via this service?

C.17.9 Send Data

Optional parameters, values, and choices supported:

- displayAttributes (physicalBaseRowNumber, physicalBaseColumnNumber, offset)
- ioCause (terminationCharReceived, charCountReached, timeout, sfTerminated)
- privateData
- privateData in the acknowledgement

Miscellaneous characteristics:

- Does the switching function support modifying the relative positions of the logical and physical displays (i.e., scrolling) via this service?

C.17.10 Send Multicast Data

Optional parameters, values, and choices supported:

- ioData
- displayAttributes (physicalBaseRowNumber, physicalBaseColumnNumber, offset)
- privateData
- privateData in the acknowledgement

Miscellaneous characteristics:

- Does the switching function support modifying the relative positions of the logical and physical displays (i.e., scrolling) via this service?

C.17.11 Start Data Path

Optional parameters, values, and choices supported:

- object (device, call)
- dataPathDirection (fromCfToObject, fromObjectToCF, biDirectional)
- dataPathType (text, voice)
- numberOfCharsToCollect
- terminationCharacter
- timeout
- privateData
- numberOfCharsToCollect in the acknowledgement
- terminationCharacter in the acknowledgement
- timeout in the acknowledgement
- privateData in the acknowledgement

C.17.12 Stop Data Path

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.17.13 Suspend Data Path

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.18 DataCollectionServList

The DataCollectionServList parameter type specifies the Data Collection services that are supported by the switching function.

C.18.1 Data Collected

Optional parameters, values, and choices supported:

- digitsDuration
- digitsPauseDuration
- toneDetected (beep, billing, busy, carrier, confirmation, dial, faxCNG, hold, howler, intrusion, modemCNG, park, recordWarning, reorder, ringback, silence, sitVC, sitIC, sitRO, sitNC, sf0, sf1, sf2, sf3, sf4, sf5, sf6, sf7, sf8, sf9, sf10, other)
- toneFrequency
- toneDuration
- tonesPauseDuration
- connectionInfo
- dcollCause (flushCharReceived, charCountReached, timeout, sfTerminated)

- privateData
- privateData in the acknowledgement

C.18.2 Data Collection Resumed

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.18.3 Data Collection Suspended

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.18.4 Resume Data Collection

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.18.5 Start Data Collection

Optional parameters, values, and choices supported:

- object (device, call)
- dataCollectionType (digits, telephonyTones)
- digitsReportingCriteria (numChars, flushChar, timeout)
- privateData
- privateData in the acknowledgement

C.18.6 Stop Data Collection

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.18.7 Suspend Data Collection

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.19 VoiceUnitServList

The VoiceUnitServList parameter type specifies the voice unit services that are supported by the switching function.

C.19.1 Concatenate Message

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.19.2 Delete Message

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.19.3 Play Message

Optional parameters, values, and choices supported:

- duration
- termination parameter, termination (durationExceeded, dtmfDigitDetected, endOfSpeechDetected, speech)
- privateData
- privateData in the acknowledgement

Miscellaneous characteristics:

- Can multiple messages be played on the same connection at the same time?

C.19.4 Query Voice Attribute

Optional parameters, values, and choices supported:

- attributeToQuery (encodingAlgorithm, samplingRate, duration, filename, currentPosition, currentSpeed, currentVolume, currentLevel, currentState)
- connection
- duration
- termination parameter, termination (durationExceeded, dtmfDigitDetected, endOfSpeechDetected, speech)
- privateData
- attribute (encodingAlgorithm (aDPCM6K, aDPCM8K, muLawPCM6k, aLawPCM6K) samplingRate, duration, filename, currentPosition, currentSpeed, currentVolAbs, currentGain, currentState) in the acknowledgement
- privateData in the acknowledgement

C.19.5 Record Message

Optional parameters, values, and choices supported:

- samplingRate
- encodingAlgorithm (aDPCM6K, aDPCM8K, muLawPCM6k, aLawPCM6K)
- maxDuration
- termination parameter, termination (durationExceeded, dtmfDigitDetected, endOfDataDetected, speechDetected)
- privateData
- privateData in the acknowledgement

C.19.6 Reposition

Optional parameters, values, and choices supported:

- periodOfReposition (startOfMessage, endOfMessage, relativePointer)
- messageToReposition
- privateData
- privateData in the acknowledgement

C.19.7 Resume

Optional parameters, values, and choices supported:

- messageToResume
- duration
- privateData
- privateData in the acknowledgement

C.19.8 Review

Optional parameters, values, and choices supported:

- periodToResume (startOfMessage, lengthOfReview)
- messageToReview
- privateData
- privateData in the acknowledgement

C.19.9 Set Voice Attribute

Optional parameters, values, and choices supported:

- currentSpeed
- currentVolume (currentVolAbs, currentVolInc)
- periodToResume (startOfMessage, lengthOfReview)
- currentGain
- message
- privateData
- privateData in the acknowledgement

C.19.10 Stop

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.19.11 Suspend

Optional parameters, values, and choices supported:

- message
- privateData
- privateData in the acknowledgement

C.19.12 Synthesize Message

Optional parameters, values, and choices supported:

- gender (male, female)
- privateData
- privateData in the acknowledgement

C.20 VoiceUnitEvtsList

The VoiceUnitEvtsList parameter type specifies the voice unit events that are supported by the switching function.

C.20.1 Play

Optional parameters, values, and choices supported:

- length

- currentPosition
- speed
- cause
- servicesPermitted
- privateData

C.20.2 Record

Optional parameters, values, and choices supported:

- length
- currentPosition
- speed
- cause
- servicesPermitted
- privateData

C.20.3 Review

Optional parameters, values, and choices supported:

- length
- currentPosition
- cause
- servicesPermitted
- privateData

C.20.4 Stop

Optional parameters, values, and choices supported:

- length
- currentPosition
- speed
- cause
- servicesPermitted
- privateData

C.20.5 Suspend Play

Optional parameters, values, and choices supported:

- length
- currentPosition
- cause
- servicesPermitted
- privateData

C.20.6 Suspend Record

Optional parameters, values, and choices supported:

- length
- currentPosition
- cause
- servicesPermitted
- privateData

C.20.7 Voice Attribute Changed

Optional parameters, values, and choices supported:

- playVolume (playVolAbs, playVolInc)
- recordingGain
- speed
- currentPosition
- cause
- privateData

C.21 CDRServList

The CDRServicesServList parameter type specifies the CDR services that are supported by the switching function.

C.21.1 Call Detail Records Notification

Optional parameters, values, and choices supported:

- cdrReason (timeout, thresholdReached, other)
- privateData
- privateData in the acknowledgement

C.21.2 Call Detail Records Report

Optional parameters, values, and choices supported:

- cdrReason (timeout, thresholdReached, other)
- recordNumber
- recordCreationTime
- callingDevice
- calledDevice
- associatedCallingDevice
- associatedcalledDevice
- networkCallingDevice
- networkCalledDevice
- callCharacteristics
- mediaCallCharacteristics
- chargedDevice (operator, nonOperator)
- recordedCall
- nodeNumber (area0, area1, area2)
- tariffTable
- connectionStart

- connectionEnd
- connectionDuration
- accessCode
- carrier
- selectedRoute
- billingIndicator (normalCharging, reverseCharging, creditCardCharging, callForwarding, callDeflection, callTransfer, other)
- chargingInfo
- suppServiceInfo (normalCall, consultationCall, transferCall, callCompletion, callForwarding, callDiversion, conferencing, intrusion, userUserInfo, other)
- reasonForTerm (normalClearing, unsuccessfulCallAttempt, abnormalTermination, callTransferred, other)
- authCode
- accountInfo
- deviceCategory
- namedDeviceTypes
- operatorDevice
- lastStoredCDRReportSent
- privateData
- privateData in the acknowledgement

C.21.3 Send Stored Call Detail Records

Optional parameters, values, and choices supported:

- timePeriod
- privateData
- privateData in the acknowledgement

C.21.4 Start Call Detail Records Transmission

Optional parameters, values, and choices supported:

- transferMode (transferAtEndOfCall, transferOnRequest, transferOnThresholdReached)
- privateData
- privateData in the acknowledgement

C.21.5 Stop Call Detail Records Transmission

Optional parameters, values, and choices supported:

- cdrTermReason (endOfDataDetected, errorDetected, thresholdReached, other)
- privateData
- privateData in the acknowledgement

Miscellaneous characteristics:

- Does the switching function generate this service request?
- Does the switching function support receiving this request?

C.22 VendorSpecificServList

The VendorSpecificServList parameter type specifies the vendor specific extension services which the switching function might request/send from the computing function or that the switching function supports from the computing function.

C.22.1 Escape Register

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.22.2 Escape Register Abort

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.22.3 Escape Register Cancel

Optional parameters, values, and choices supported:

- privateData
- privateData in the acknowledgement

C.22.4 Escape

Optional parameters, values, and choices supported:

- privateData in the acknowledgement

Miscellaneous Characteristics:

- Does the switching function generate this service request?
- Does the switching function support receiving this service request?

C.22.5 Private Data Version Selection

Optional parameters, values, and choices supported:

- privateData in the acknowledgement

C.23 VendorSpecificEvtsList

The VendorSpecificEvtsList parameter type specifies the vendor specific events that are supported by the switching function.

C.23.1 Private Event

None.

Annex D
(informative)

Connection State Transition Examples

This annex contains examples that describes the connection state transitions specified in 6.1.5, “Connection”.

Table D-1 Connection State Transition Call Flow Examples

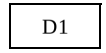
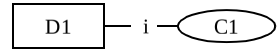
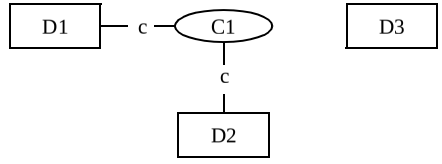
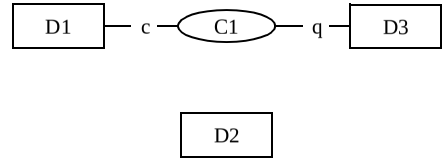
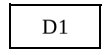
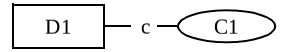
State Change	Action	Initial State	Final State	Comments
(D1) Null -> Initiated	A device goes off-hook or the computing function issues a Make Call service and the calling device is prompted.			
(D3) Null -> Queued	Device D2 parks the call to another device (D3).			
(D1) Null -> Connected	The computing function issues a Make Call service request. The switching function sends the Originated event for the calling device.			The calling device is an Auto Answer device therefore no Service Initiated event is seen prior to the Originated event.
(D2) Null -> Fail	Device D1 dials Device D2. Device D2 is currently active in another call and can not accept D1's call. D1 hears a busy tone.			
(D2) Null -> Alerting	Switching function alerts a device. This could be the result of a Make Call service and the called device is being alerted.			The switching function could alert a device that originates outside the device's switching domain.

Table D-1 Connection State Transition Call Flow Examples (continued)

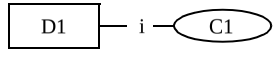
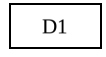
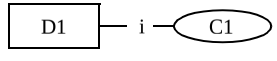
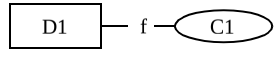
State Change	Action	Initial State	Final State	Comments
(D2) Null -> Null	Device D1 dials Device D2. Device D2 is busy on another call and cannot accept D1's call. The call is immediately forwarded and moves to the new device without ever creating a connection at D2. To report this, a Diverted event shall be based on a Null->Null transition.			
(D1) Initiated -> Null	A device that was taken off-hook is placed back on-hook before any actions are taken.			
(D1) Initiated -> Fail	A device that was taken off-hook is left off-hook without entering any digits. After a time-out period, the device receives the reorder tone and the computing function receives the Failed event.			

Table D-1 Connection State Transition Call Flow Examples (continued)

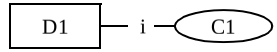
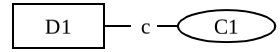
State Change	Action	Initial State	Final State	Comments
(D1) Initiated -> Connected	A device goes off-hook and finishes dialling. A call is originated to another device and the computing function receives an Originated event.			
(D1) Initiated -> Alerting	Device D1 is the calling device in a predictive call. Before any activity at D2, the D1C1 connection goes to initiated. After D2 is connected into the call, D1 is alerted.			
(D1) Initiated -> Queued	Device D1 is the calling device in a predictive call. Before any activity at D2, the D1C1 connection goes to initiated. After D2 is connected into the call, the call is queued at D1 (D1 is busy on another call).			

Table D-1 Connection State Transition Call Flow Examples (continued)

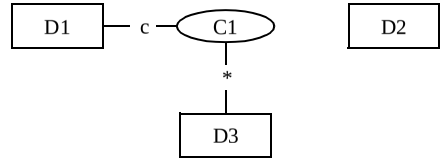
State Change	Action	Initial State	Final State	Comments
(D2) Queued -> Null	A calling device can be queued to an ACD device or queued due to a Campon situation. The call moves from the queued device to a device that accepts the call. The computing function receives a Diverted event.			
(D2) Queued -> Alerting	A device that has a camped on call clears from its active call and is alerted by the camped on call.			
(D2) Queued -> Fail	A call is queued to an ACD device and then ACD becomes unstaffed with no alternate routeing for queued calls on an unstaffed condition.			
(D2) Queued -> Hold	An appearance of a shared bridged device configuration that is in the inactive mode has been placed on hold as a result of actions by another appearance in the device configuration.			

Table D-1 Connection State Transition Call Flow Examples (continued)

State Change	Action	Initial State	Final State	Comments
(D2) Queued -> Connected	A call has been parked at device D2. The computing function successfully issues the Answer Call service for device D2.			
(D1) Connected -> Null	A device in a two-party call manually goes on-hook or a Clear Connection service is successfully invoked on behalf of this device.			
(D1) Connected -> Fail	A device that was in a two-party call stays off-hook after the other device goes on-hook. After a time-out period, the device receives the blocked tone and the computing function receives the Failed event.			
(D1) Connected -> Hold	A device in a two-party call manually places the other device on hold or the Hold Call service is successfully executed.			Device D1 is the device placing the call on hold.

Table D-1 Connection State Transition Call Flow Examples (continued)

State Change	Action	Initial State	Final State	Comments
(D2) Connected -> Queued	In a shared bridged configuration, Device D2 goes on-hook while another appearance in the bridged configuration remains connected to the call.			
(D2) Hold -> Alerting	A held or consulted call returns to a device because of a time-out.			
(D1) Hold -> Connected	A device that has previously placed another device on hold manually retrieves the held device or the Reconnect or Retrieve Call service is successfully executed.			Device D1 is the device that retrieves the call.
(D1) Hold -> Null	A device that has previously placed another device on hold manually goes on-hook or the Clear Connection service is successfully executed for the holding device.			

Table D-1 Connection State Transition Call Flow Examples (continued)

State Change	Action	Initial State	Final State	Comments
(D2) Hold -> Queued	A held appearance of a shared bridged device configuration is retrieved from hold and is placed in the inactive mode as a result of actions by another appearance in the device configuration.			
(D2) Hold -> Fail	A device that has previously placed another device on hold has exceeded the holding timer and has transitioned to the Fail state.			
(D2) Fail -> Connected	A device intrudes into a call that first failed because the called device was busy.			
(D2) Fail -> Alerting	A called device that was busy begins to alert.			
(D2) Fail -> Hold	An exclusive-bridged appearance has placed a call on hold and the other appearances which were blocked from use, transition to the hold state as well.			

Table D-1 Connection State Transition Call Flow Examples (continued)

State Change	Action	Initial State	Final State	Comments
(D2) Fail -> Queued	A called device is busy. The calling device camps on to the called device.			
(D2) Fail -> Null	A device is busy. The calling device goes back on-hook.			
(D2) Alerting -> Null	A call is delivered to a device. Before the called device answers, the calling device goes on-hook.			
(D2) Alerting -> Queued	A call is delivered to an ACD device. No agents are available so the call queues at the ACD device.			
(D2) Alerting -> Connected	A call delivered to a device. The called device either manually answers the call or the Answer Call service is used.			
(D2) Alerting -> Alerted	A call has been offered to a device. After a timeout the call is delivered to the same device.			

Table D-1 Connection State Transition Call Flow Examples (continued)

State Change	Action	Initial State	Final State	Comments
(D2) Alerting -> Fail	A call is alerting a device. That device is removed from service and there is no coverage or redirection criteria for this situation.			

Printed copies can be ordered from:

ECMA
114 Rue du Rhône
CH-1204 Geneva
Switzerland

Fax: +41 22 849.60.01
Internet: documents@ecma.ch

Files can be downloaded from our FTP site, **ftp.ecma.ch**, logging in as **anonymous** and giving your E-mail address as **password**. This Standard is available from library **ECMA-ST** as an Acrobat PDF file (file E269-PDF.PDF). File E269-EXP.TXT gives a short presentation of the Standard.

Our web site, http://www.ecma.ch, gives full information on ECMA, ECMA activities, ECMA Standards and Technical Reports.

ECMA

114 Rue du Rhône
CH-1204 Geneva
Switzerland

This Standard ECMA-269 is available free of charge in printed form and as a file.

See inside cover page for instructions