Standard ECMA-269
3rd Edition - December 1998

ECMA

Standardizing Information and Communication Systems

Services for Computer Supported Telecommunications Applications (CSTA) Phase III

Volume 2

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 2

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

17 Call Control Services & Events

This clause describes the Call Control features of this Standard. It includes:

- Call Control services
- Call Control events

For a description of fundamental concepts, such as connection states, please refer to 6.1.5 beginning on page 31.

17.1 Services

Table 17-1 Call Control Services Summary

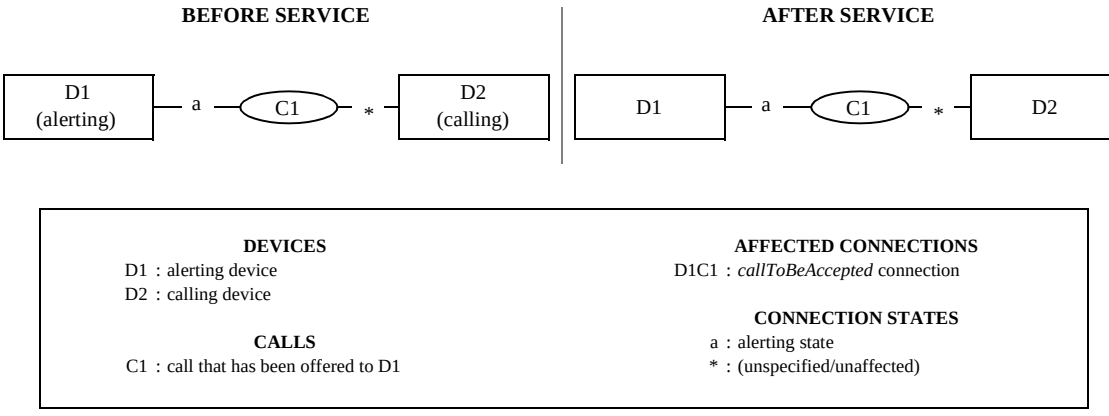
Call Control Service	Description	Pg.
17.1.1 Accept Call	Causes an offered call to transition to the Ringing or Entering Distribution mode of the Alerting state.	186
17.1.2 Alternate Call	Places an existing call on hold and then retrieves a previously held or alerting call at the same device.	188
17.1.3 Answer Call	Answers a call that is ringing, queued, or being offered to a device.	191
17.1.4 Call Back Call-Related	Allows a computing function to request that an originally called device return a call to the original calling device.	193
17.1.5 Call Back Message Call-Related	Allows a computing function to instruct the switching function to leave a pre-defined message requesting that the called device call the calling device.	196
17.1.6 Camp On Call	Queues a call at a busy device until the device becomes available.	199
17.1.7 Clear Call	Releases all of the devices associated with the specified call.	201
17.1.8 Clear Connection	Releases a specific device from a call.	204
17.1.9 Conference Call	Provides a conference of an existing held call and another active call at a conferencing device. The two calls are merged into a single call at the conferencing device.	208
17.1.10 Consultation Call	Places an existing active call at a device on hold and initiates a new call from the same device.	211
17.1.11 Deflect Call	Deflects a call to another device.	217
17.1.12 Dial Digits	Dials a digit sequence for a call that has already been initiated.	220
17.1.13 Directed Pickup Call	Picks a specified call. (Moves and connects a specified alerting or queued call.)	223
17.1.14 Group Pickup Call	Picks a call from a specified pick group. (Moves and connects any alerting call in a pick group to another device.)	226
17.1.15 Hold Call	Places a specific connection on hold.	229
17.1.16 Intrude Call	Allows a computing function to add the calling device to a call at a busy called device.	231
17.1.17 Join Call	Allows a computing function to request, on behalf of a device, that the device be joined into an existing call.	235
17.1.18 Make Call	Establishes a call between two devices.	239
17.1.19 Make Predictive Call	Establishes a call between two devices. The calling device is presented with the call only after the called device is alerted or has answered the call.	245
17.1.20 Park Call	Parks a call at a specified device. (Moves and queues a connected call to another device).	250
17.1.21 Reconnect Call	Clears an existing connection and then connects a previously held connection at the same device.	253
17.1.22 Retrieve Call	Connects to a call that had previously been placed on hold.	255
17.1.23 Single Step Conference Call	Adds a device to an existing call.	257
17.1.24 Single Step Transfer Call	Replaces a device in an existing call with another device.	261
17.1.25 Transfer Call	Transfers a held call to the consulted party.	264

17.1.1 Accept Call

C → S

The Accept Call service causes an offered call to transit from the offered mode to the Ringing or Entering Distribution mode of the alerting state.

Figure 17-1 Accept Call Service



17.1.1.1 Service Request

Table 17-2 Accept Call—Service Request

Parameter Name	Type	M/ O/C	Description
callToBeAccepted	ConnectionID	M	Specifies the connection to be accepted.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
userData	UserData	O	Specifies the user data to be sent to parties in the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.1.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.1.2.1 Positive Acknowledgement

Table 17-3 Accept Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.1.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.1.3 Operational Model

17.1.1.3.1 Connection State Transitions

Table 17-4 Accept Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (callToBeAccepted)	Alerting (Offered mode only)	Alerting (Ringing or Entering Distribution mode)
other connections (i.e., D2C1)	(Unspecified)	(Unaffected; no transition due to this service).

17.1.1.3.2 Device-Type Monitoring Event Sequences

Table 17-5 Accept Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (Alerting device)	D1C1 (callToBeAccepted)	Delivered	Normal, Entering Distribution
D2 (Calling device or any other devices in call C1)	D1C1 (callToBeAccepted)	Delivered	Normal, Entering Distribution

17.1.1.3.3 Call-Type Monitoring Event Sequences

Table 17-6 Accept Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D1C1 (callToBeAccepted)	Delivered	Normal, Entering Distribution

17.1.1.3.4 Functional Requirements

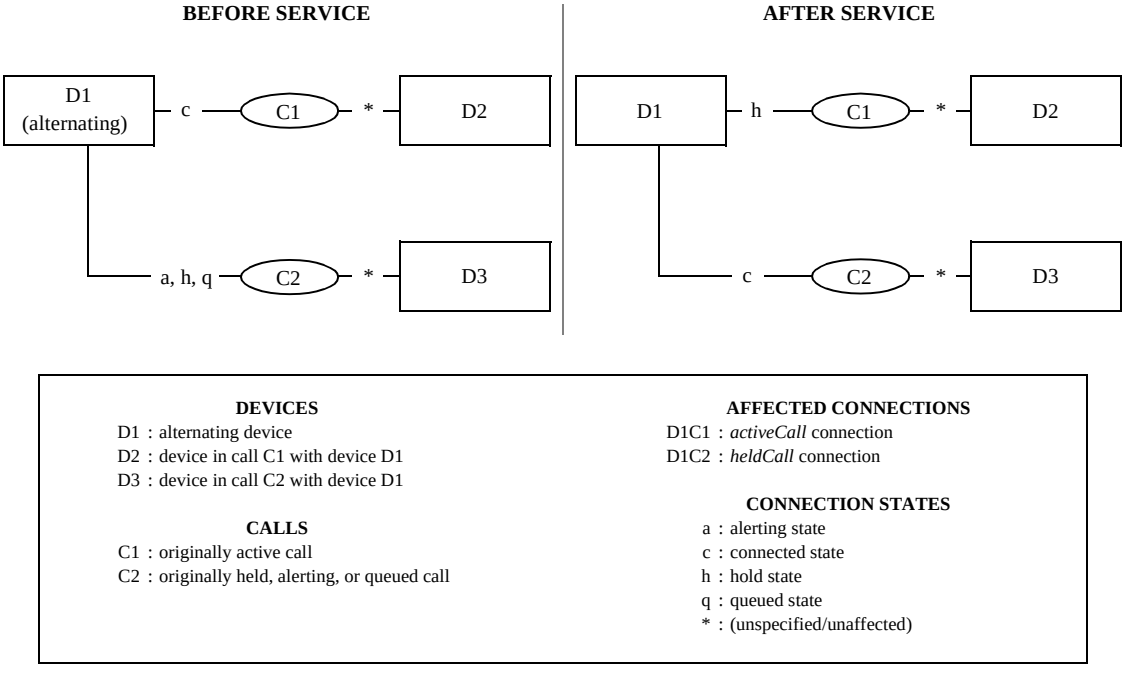
1. If the computing function wants to clear an active call prior to accepting an alerting call for a device, it shall first issue a Clear Connection service for the active call and then issue the Accept Call service for the alerting call.

17.1.2 Alternate Call

C → S

The Alternate Call service places an existing active call on hold and then retrieves a previously held call. This service is also used to place an active call on hold and then connect to an alerting or queued call at the same device (i.e., to answer a call-waiting call).

Figure 17-2 Alternate Call Service



17.1.2.1 Service Request

Table 17-7 Alternate Call—Service Request

Parameter Name	Type	M/ O/C	Description
heldCall	ConnectionID	M	Specifies the held connection for the alternating device.
activeCall	ConnectionID	M	Specifies the active connection for the alternating device.
connectionReservation	Boolean	O	Specifies that the media stream channel(s) associated with the call being placed on hold be reserved for reuse at a later time. The complete set of possible values is: • True - channel(s) is to be reserved. • False - channel(s) is not to be reserved (default).

Table 17-7 Alternate Call—Service Request (continued)

Parameter Name	Type	M/ O/C	Description
consultOptions	Enumerated	C	This parameter indicates the potential actions following the Alternate Call so that certain facilities can be allocated prior to a Transfer or Conference. The complete set of possible values is: • Consult Only • Transfer Only • Conference Only • Unrestricted (default) If the switching function supports this parameter, the computing function shall supply one of the values supported. If the switching function does not support this parameter, the implicit value is Unrestricted. This parameter does not indicate a restriction or otherwise affect the switching function's capabilities in any way with respect to services other than Transfer or Conference.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information.

17.1.2.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.2.2.1 Positive Acknowledgement

Table 17-8 Alternate Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.2.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, "ErrorValue", on page 88.

17.1.2.3 Operational Model

17.1.2.3.1 Connection State Transitions

Table 17-9 Alternate Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (activeCall)	Connected	Hold
D1C2 (heldCall)	Hold, Alerting, or Queued	Connected
other connections (i.e., D2C1, D3C2)	(Unspecified)	(Unaffected; no transition due to this service).

17.1.2.3.2 Device-Type Monitoring Event Sequences

Sequence of events using call-type monitoring is defined in Table 17-10 for the following cases:

- Case A: Call C2 is on Hold at device D1 (Alternating Device)

- Case B: Call C2 is Alerting or Queued at device D1 (Alternating Device)

Table 17-10 Alternate Call—Device-Type Monitoring Event Sequences (Case A and B)

Monitored Device	Connection	Event	Event Cause
D1 (Alternating device)	D1C1 (activeCall)	Held	Transfer, Conference, Alternate, or Normal
	D1C2 (heldCall)	Retrieved (Case A)	Transfer, Conference, Alternate, or Normal
		Established (Case B)	
D2 (or any other devices in conference with D2)	D1C1 (activeCall)	Held	Transfer, Conference, Alternate, or Normal
D3 (or any other devices in conference with D3)	D1C2 (heldCall)	Retrieved (Case A)	Transfer, Conference, Alternate, or Normal
		Established (Case B)	

17.1.2.3.3 Call-Type Monitoring Event Sequences

Sequence of events using call-type monitoring is defined in Table 17-11 for the following cases:

- Case A: Call C2 is on Hold at device D1 (Alternating Device)
- Case B: Call C2 is Alerting or Queued at device D1 (Alternating Device)

Table 17-11 Alternate Call—Call-Type Monitoring Event Sequences (Case A and B)

Monitored Call	Connection	Event	Event Cause
C1 (originally active call)	D1C1 (activeCall)	Held	Transfer, Conference, Alternate, or Normal
C2 (originally held, alerting, or queued call)	D1C2 (heldCall)	Retrieved (Case A)	Transfer, Conference, Alternate, or Normal
		Established (Case B)	

17.1.2.3.4 Functional Requirements

1. The Alternate Call service shall not be used to put an alerting call on hold and re-connect to another call that is on hold.
2. This service is a multiple step service that is equivalent to the computing function issuing the Hold Call service for the activeCall ConnectionID and then issuing one of the following services:
   a. a Retrieve Call service for the heldCall ConnectionID.
   b. an Answer Call service for the heldCall ConnectionID.
   c. an Accept Call service for the heldCall ConnectionID.
3. The consultOptions parameter indicates the potential action following the Alternate Call service so that certain facilities can be allocated if a transfer or conference is desired. If the switching function supports the consultOptions parameter, the computing function shall provide one of the supported values of this parameter obtained through the capability exchange services. If the switching function requires preallocation of resources to support transfer or conference but is unable to allocate the resources required to support the indicated usage at this time, it shall reject the Alternate Call service request. This parameter does not indicate a restriction or otherwise affect the switching function’s capabilities in any way with respect to services other than Transfer or Conference.
4. If all appearances of a shared bridged device configuration are in the hold state and the heldCall parameter contains an appearance’s connection ID in the call, then the other appearances in the device configuration will return to the inactive mode (Queued state, Bridged Events).

17.1.3 Answer Call

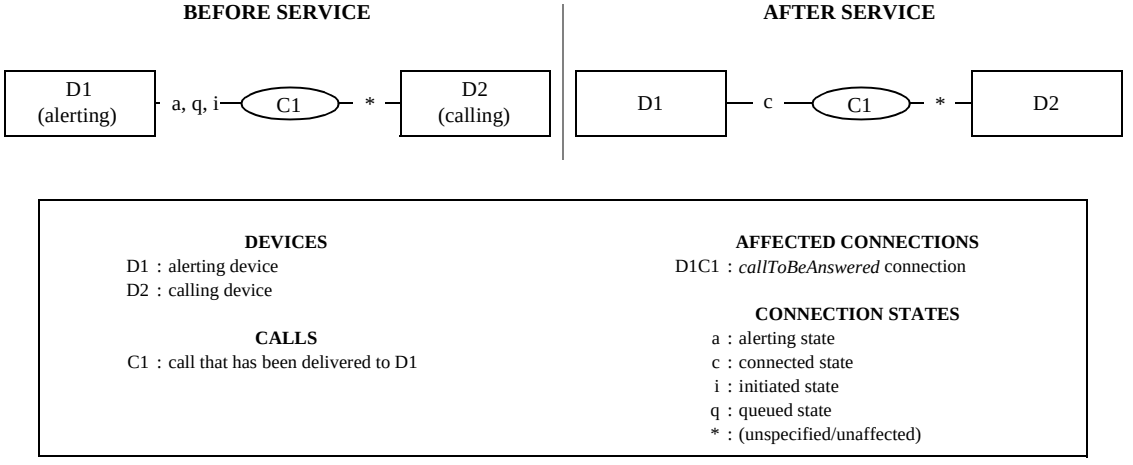
C → S

The Answer Call service connects an alerting or queued call.

This service is typically associated with devices that have attached speakerphone units and headset telephones to connect to a call via hands-free operation. For example, when the call is answered, one of the following actions may occur:

- If the specified device has a speaker and a microphone, the speaker and microphone are turned on.
- If the specified device only has a speaker, the speaker is turned on. The handset shall be picked up in order to have a two way conversation.
- If there is no speaker, then the handset shall be picked up in order to have a two-way conversation.
- If the specified device has a headset, the headset is turned on.

Figure 17-3 Answer Call Service



Note that D1 may also be the calling device (e.g. Make Predictive Call).

17.1.3.1 Service Request

Table 17-12 Answer Call—Service Request

Parameter Name	Type	M/ O/C	Description
callToBeAnswered	ConnectionID	M	Specifies the connection to be answered.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
userData	UserData	O	Specifies the user data to be sent to parties in the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.3.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.3.2.1 Positive Acknowledgement

Table 17-13 Answer Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.3.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.3.3 Operational Model

17.1.3.3.1 Connection State Transitions

Table 17-14 Answer Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (callToBeAnswered)	Alerting, Initiated, or Queued	Connected
other connections (i.e., D2C1)	(Unspecified)	(Unaffected; no transition due to this service).

17.1.3.3.2 Device-Type Monitoring Event Sequences

Table 17-15 Answer Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (Alerting device)	D1C1 (callToBeAnswered)	Established	Normal
D2 (Calling device or any other devices in conference with D2)	D1C1 (callToBeAnswered)	Established	Normal

17.1.3.3.3 Call-Type Monitoring Event Sequences

Table 17-16 Answer Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D1C1 (callToBeAnswered)	Established	Normal

17.1.3.3.4 Functional Requirements

1. If the computing function wants to clear an active call prior to answering an alerting or queued call for a device, it shall first issue a Clear Connection service for the active call and then issue the Answer Call service for the alerting or queued call.

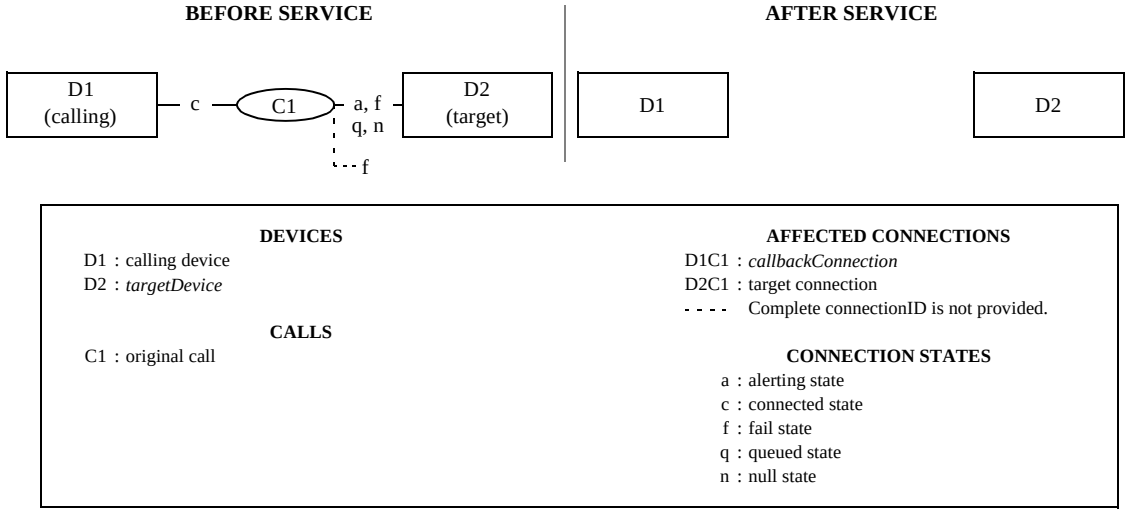
17.1.4 Call Back Call-Related

C → S

The Call Back Call-Related service allows a computing function to request that the calling device retry the call to the called device when the called device is in an appropriate state to accept the call.

As an example, the service might be used when a called device was busy so that the call is reattempted when the device becomes free.

Figure 17-4 Call Back Call-Related Service



Refer to 6.8.2, “Connection Failure”, on page 53 for a complete description of partial connection IDs.

17.1.4.1 Service Request

Table 17-17 Call Back Call-Related —Service Request

Parameter Name	Type	M/ O/C	Description
callbackConnection	ConnectionID	M	Specifies the call back connection at the calling device.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (Priority call, for example) to be associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values. If the supported characteristics cannot be honoured, the switching function shall reject the service request.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.4.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.4.2.1 Positive Acknowledgement

Table 17-18 Call Back Call-Related—Positive Acknowledgement

Parameter Name	Type	M/O/C	Description
targetDevice	DeviceID	C	Specifies the deviceID of the device that the call back was initiated for. This parameter is mandatory if the switching function supports the Cancel Call Back service, otherwise it is optional.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.4.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.4.3 Operational Model

17.1.4.3.1 Connection State Transitions

Table 17-19 Call Back Call-Related —Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (callback)	Connected	Null, Fail (see item #1)
D2C1 (called)	Fail, Alerting, Queued, or Null (e.g., null if call was forwarded or deflected from D2)	Null

1. A device may transition to the Fail state prior to Null if the device stays off-hook and receives busy or blocked tone.
2. In the case where a call is forwarded from a called device (busy forwarding, for example), it is switching function dependent if the call back request is placed on the called device or the device that the call was forwarded to.

17.1.4.3.2 Device-Type Monitoring Event Sequences

Table 17-20 Call Back Call-Related—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (calling device)	D2C1 (called)	Connection Cleared (See items #2 and #3)	Call Back or Normal Clearing
	D1C1 (callback)	Connection Cleared (see items #1 and #2)	Call Back or Normal Clearing
		Call Back	
D2 (called device)	D2C1 (called)	Connection Cleared (See items #2 and #3)	Call Back or Normal Clearing

17.1.4.3.3 Call-Type Monitoring Event Sequences

Table 17-21 Call Back Call-Related—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D2C1 (called)	Connection Cleared (see items #2 and #3)	Call Back or Normal Clearing
	D1C1 (callback)	Connection Cleared (see items #1 and #2)	Call Back or Normal Clearing
	(D1C1)	Call Cleared	Call Back or Normal Clearing

1. If a device stays off-hook and receives busy or blocked tone, the switching function sends the Failed event (event cause of Blocked or Busy) to indicate the status of the device, followed by a Connection Cleared event.
2. The exact sequence of the Connection Cleared events for this service is not specified. This is due to the fact that when multiple devices are removed from a call as a result of the service, each switching function may process the clearing of the devices in a different order.
3. This event is sent only if the connection associated with the called device was created.
4. The event sequence if the call is forwarded or deflected from D2 is not shown.

17.1.4.3.4 Functional Requirements

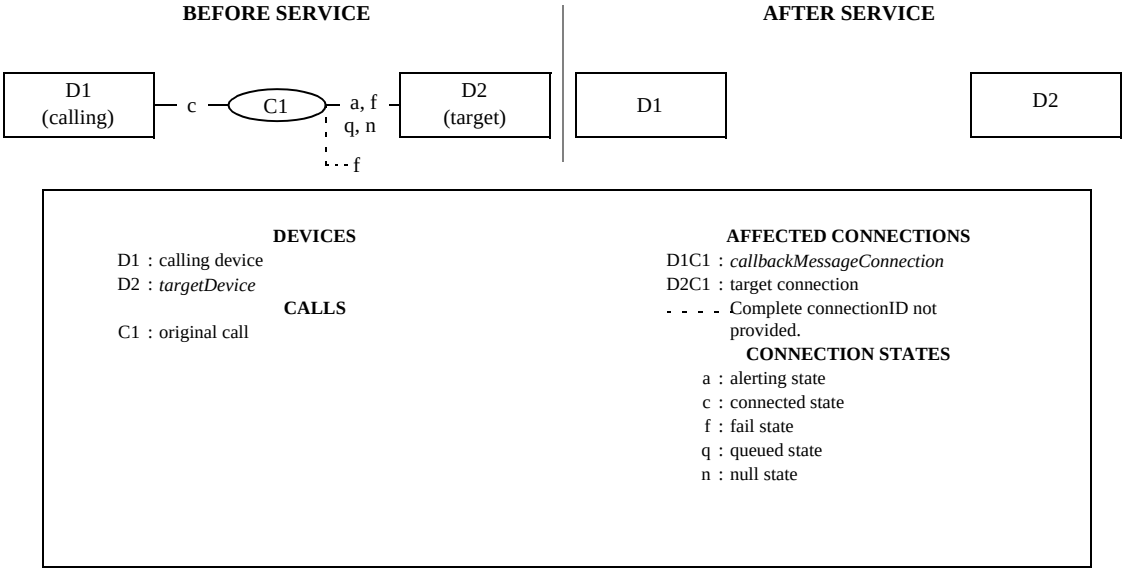
1. Call Back Call-Related is similar to the Camp On Call service. However for the Call Back Call-Related service the original call is cleared, but for the Camp On Call service, the call is queued.
2. Only one Call Back service request (Call-Related or Non-Call-Related) can be outstanding for any calling and called device pair. It is a switching function option (as indicated by the capability exchange services) if additional Call Back service requests (Call-Related or Non-Call-Related) for that pair result in a positive or negative acknowledgement from the switching function.
3. To cancel a Call Back (Call-Related or Non-Call-Related), the computing function shall issue the Cancel Call Back service, alternatively the Call Back should be manually canceled.

17.1.5 Call Back Message Call-Related

C ⟶ S

The Call Back Message Call-Related service allows a computing function to request that the switching function leave a pre-defined message requesting that the called device call the calling device. For example, the called device may have been busy when called.

Figure 17-5 Call Back Message Call-Related Service



Refer to 6.8.2, “Connection Failure”, on page 53 for a complete description of partial connection IDs.

17.1.5.1 Service Request

Table 17-22 Call Back Message Call-Related—Service Request

Parameter Name	Type	M/ O/C	Description
callbackMessageConnection	ConnectionID	M	Specifies the call back message connection at the calling device.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.5.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.5.2.1 Positive Acknowledgement

Table 17-23 Call Back Message Call-Related—Positive Acknowledgement

Parameter Name	Type	M/O/ C	Description
targetDevice	deviceID	C	Specifies the deviceID of the device that the call back message was left for. This parameter is mandatory if the switching function supports the Cancel Call Back Message service, otherwise it is optional.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.5.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.5.3 Operational Model

17.1.5.3.1 Connection State Transitions

Table 17-24 Call Back Message Call-Related—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (callbackMessage)	Connected	Null, Fail (see item #1)
D2C1 (called)	Fail, Alerting, Queued, or Null (e.g., null if call was forwarded, deflected from D2)	Null

1. A device may transition to the Fail state prior to Null if the device stays offhook and receives busy or blocked tone.
2. In the case where a call is forwarded from a called device (busy forwarding, for example), it is switching function dependent if the call back request is placed on the called device or the device that the call was forwarded to.

17.1.5.3.2 Device-Type Monitoring Event Sequences

Table 17-25 Call Back Message Call-Related—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (calling device)	D2C1 (called)	Connection Cleared (see items #3 and #4)	Call Back or Normal Clearing
	D1C1 (calling)	Connection Cleared (see items #2 and #4)	Call Back or Normal Clearing
		Call Back Message	
D2 (called device)	D2C1 (called)	Connection Cleared (see items #3 and #4)	Call Back or Normal Clearing

17.1.5.3.3 Call-Type Monitoring Event Sequences

Table 17-26 Call Back Message Call-Related—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D2C1 (called)	Connection Cleared (see items #3 and #4)	Call Back or Normal Clearing
	D1C1 (calling)	Connection Cleared (see items #2 and #4)	Call Back or Normal Clearing
	(D1C1)	Call Cleared	Call Back or Normal Clearing

1. The event sequence if the call is forwarded or deflected from D2 is not shown.
2. If a device stays off-hook and receives busy or blocked tone, the switching function sends the Failed event (event cause of Blocked or Busy) to indicate the status of the device, followed by a Connection Cleared event.
3. This event is sent only if the connection associated with the called device was created.
4. The exact sequence of the Connection Cleared events for this service is not specified. This is due to the fact that when multiple devices are removed from a call as a result of the service, each switching function may process the clearing of the devices in a different order.

17.1.5.3.4 Functional Requirements

1. Only one Call Back Message service request (Call-Related or Non-Call-Related) can be outstanding for any calling and called device pair. It is a switching function option (as indicated by the capability exchange

services) if additional Call Back Message service requests (Call-Related or Non-Call-Related) for that pair result in a positive or negative acknowledgement from the switching function.

2. To cancel a Call Back Message (Call-Related or Non-Call-Related), the computing function shall issue the Cancel Call Back Message service, alternatively the Call Back Message should be manually canceled.

3. The Call Back Message service (Call-Related or Non-Call-Related) differs from the Call Back service in that with the Call Back Message service, the calling device will not call back the called device. Instead, this service leaves a message at the called device.

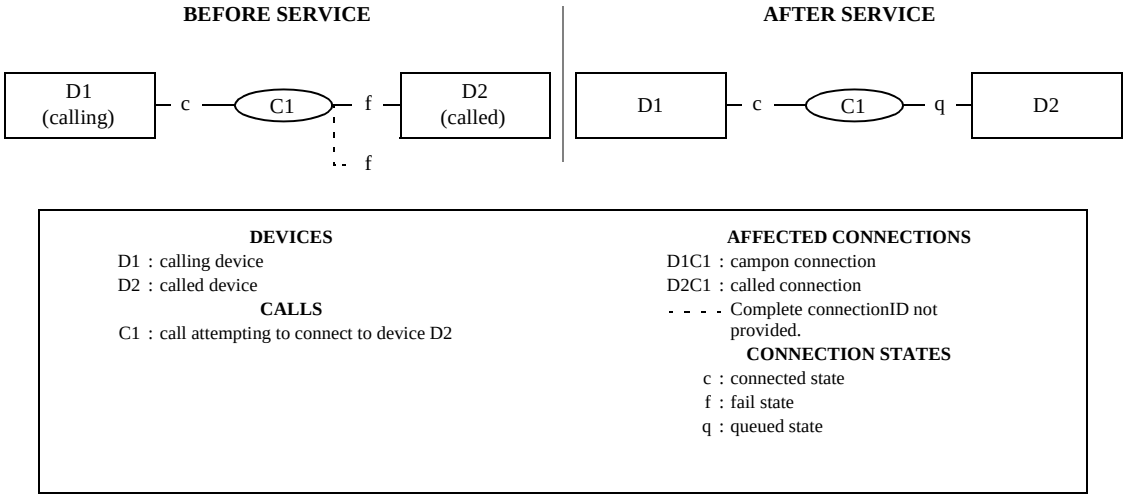
4. The switching function defines the message left at the called device. A computing function cannot use the service to specify the message content or how the switching function will notify the user (e.g., text message, voice message, indicator only).

17.1.6 Camp On Call

C → S

The Camp On Call service allows the computing function to queue a call for a device (that typically is busy) until that device becomes available (after finishing a current call or any previously queued calls, for example).

Figure 17-6 Camp On Call Service



Refer to 6.8.2, “Connection Failure”, on page 53 for a complete description of partial connection IDs.

17.1.6.1 Service Request

Table 17-27 Camp On Call—Service Request

Parameter Name	Type	M/ O/C	Description
camponConnection	ConnectionID	M	Specifies the connection of the calling device.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.6.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.6.2.1 Positive Acknowledgement

Table 17-28 Camp On Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.6.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.6.3 Operational Model

17.1.6.3.1 Connection State Transitions

Table 17-29 Camp On Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (campon)	Connected	Connected
D2C1 (called)	Fail	Queued

17.1.6.3.2 Device-Type Monitoring Event Sequences

Table 17-30 Camp On Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (Calling device)	D2C1 (campon)	Queued	Camp On or Camp On Trunks
D2 (Called device)	D2C1 (called)	Queued	Camp On or Camp On Trunks

17.1.6.3.3 Call-Type Monitoring Event Sequences

Table 17-31 Camp On Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D2C1 (called)	Queued	Camp On or Camp On Trunks

17.1.6.3.4 Functional Requirements

1. Only one Camp On Call feature can be active for any calling and called device pair. It is a switching function option (as indicated by the capability exchange services) if additional Camp On Call service requests for that pair result in a positive or negative acknowledgement from the switching function.
2. To cancel a Camp On Call service, the computing function can either:
   - Issue the Clear Connection service or Clear Call service with the campon connection.
   - Have the calling device go on-hook.

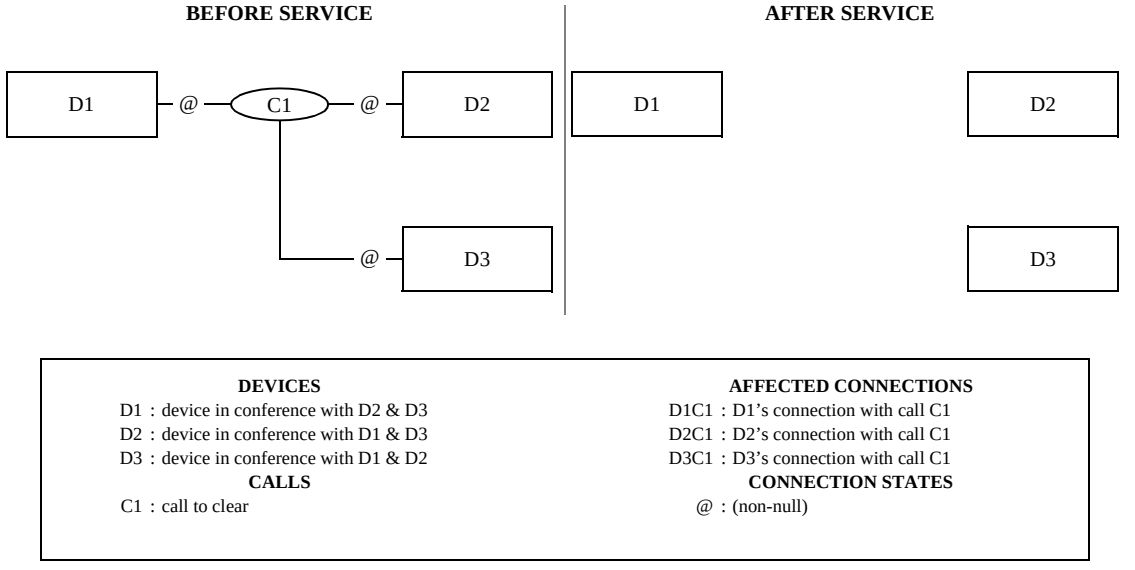
   Note that if the Camp On Call service is successfully cancelled, normal call progress messages of Connection Cleared (with an event cause of Normal Clearing) will be generated.

17.1.7 Clear Call

C → S

The Clear Call service releases all devices from an existing call. In the case of a conference call, this results in all devices in the conference call being released from the call.

Figure 17-7 Clear Call Service



The callToBeCleared connection may be any connection in the call (i.e., D1C1, D2C1, D3C1).

Note that some connections in the call may transit to the Failed state prior to the Null state. See "Connection State Transitions" on page 202.

17.1.7.1 Service Request

Table 17-32 Clear Call—Service Request

Parameter Name	Type	M/ O/C	Description
callToBeCleared [1]	ConnectionID	M	Specifies the connection to be cleared.
userData	UserData	O	Specifies the user data to be sent to parties in the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

1. The deviceID in the callToBeCleared ConnectionID may be omitted for this service.

17.1.7.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.7.2.1 Positive Acknowledgement

Table 17-33 Clear Call— Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.7.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.7.3 Operational Model

17.1.7.3.1 Connection State Transitions

Table 17-34 Clear Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1	(Any non-Null state)	Null, Failed (see item # 1)
D2C1	(Any non-Null state)	Null, Failed (see item # 1)
D3C1	(Any non-Null state)	Null, Failed (see item # 1)

1. A connection may transition to the Failed state prior to Null if the device stays off-hook and receives busy or blocked tone. The time between the state transition of Failed --> Null is device and switching function specific and the time may be substantial. Note that the service is completed when all connections transit to either the Null or the Failed state.

17.1.7.3.2 Device-Type Monitoring Event Sequences

Table 17-35 Clear Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1	D1C1	Failed (Optional) (see item #3)	Blocked, or Normal or Busy
		Connection Cleared (see items #1 and #2)	Normal Clearing
D2	D2C1	Failed (Optional) (see item #3)	Blocked, or Normal or Busy
		Connection Cleared (see items #1 and #2)	Normal Clearing
D3 (and any other devices in call C1)	D3C1	Failed (Optional) (see item #3)	Blocked, or Normal or Busy
		Connection Cleared (see items #1 and #2)	Normal Clearing

17.1.7.3.3 Call-Type Monitoring Event Sequences

Table 17-36 Clear Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D1C1	Failed (Optional) (see item #3)	Blocked, or Normal, or Busy
		Connection Cleared (see item #2)	Normal Clearing
	D2C1	Failed (Optional) (see item #3)	Blocked, or Normal, or Busy
		Connection Cleared (see item #2)	Normal Clearing
	D3C1	Failed (Optional) (see item #3)	Blocked, or Normal, or Busy
		Connection Cleared (see item #2)	Normal Clearing
	(D3C1)	Call Cleared, once the last device has left the call. (Note that this event is only sent for call-type monitors.)	Normal Clearing

1. For device-type monitors, it is always guaranteed that the Connection Cleared event associated with the monitored device is provided, and that this is the last Connection Cleared event for the cleared call over the monitor. Depending on how the switching function processes the service request, Connection Cleared events for the other devices in the call may precede this last Connection Cleared event.
2. The exact sequence of the Connection Cleared events for this service is not specified. This is due to the fact that when multiple devices are removed from a call as a result of the service, each switching function may process the clearing of those devices in a different order.
3. If a device stays off-hook and receives busy or blocked tone, the switching function sends the Failed event (event cause of Blocked or Busy) to indicate the status of the device, followed by a Connection Cleared event. The time between the generation of the Failed event and the Connection Cleared event is device and switching function specific and the time may be substantial. As for generation of the Failed event for a given monitor and device in the call, it follows the same conditions as the Connection Cleared events (if the event is supported by the switching function): No events reporting connection transitions for other devices in the call will be sent. However, the Connection Cleared event for the given monitor will be sent (and will be the last event associated with the cleared call sent over that monitor) when the connection transits to Null.

17.1.7.3.4 Functional Requirements

1. The Clear Call service shall only affect the callToBeCleared ConnectionID’s call. Other calls that exist at the callToBeCleared ConnectionID’s device remain unaffected.
2. The connection ID provided in the request may consist of only a callID portion. This is an exception to the rule of not providing deviceIDs in connectionIDs of service requests (see 6.1.5, “Connection”, on page 31).

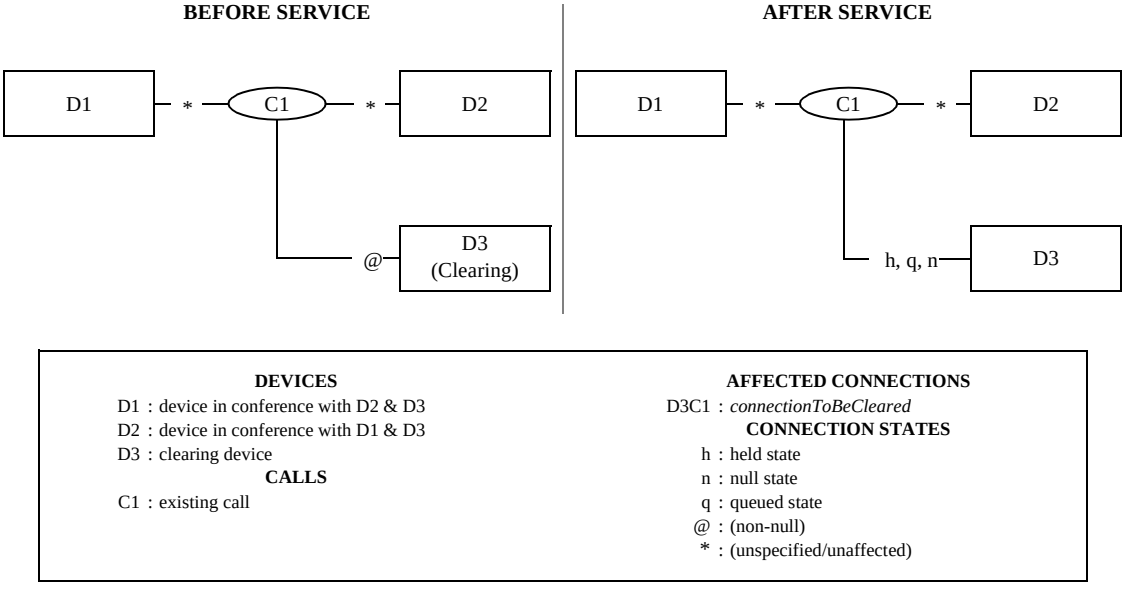
17.1.8 Clear Connection

C ⟶ S

The Clear Connection service releases a specific device from a call. In the case of a two-party call, this may result in the call being torn down. In the case of a conference call, this results in the specific party being removed from the conference. This service can also be used to inactivate a bridged appearance.

The Connection ID provided in the request is released.

Figure 17-8 Clear Connection Service



17.1.8.1 Service Request

Table 17-37 Clear Connection—Service Request

Parameter Name	Type	M/ O/C	Description
connectionToBeCleared	ConnectionID	M	Specifies the connection to be cleared.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
userData	UserData	O	Specifies the user data to be sent to parties in the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.8.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.8.2.1 Positive Acknowledgement

Table 17-38 Clear Connection—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.8.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.8.3 Operational Model

17.1.8.3.1 Connection State Transitions

Table 17-39 Clear Connection—Connection State Transitions

Connection	Initial State (Required)	Final State
D3C1 (connectionToBe-Cleared)	(Any non-Null state)	Failed (See item #1), Held (See item #4), Null, Queued (See item #5)
other Connections in call C1 (i.e., D1C1, D2C1)	(Unspecified)	Null, Failed (See item #3).

1. A connection may transition to the Failed state prior to Null if the device stays off-hook and receives busy or blocked tone. The time between the state transition of Failed -> Null is device and switching function specific and the time may be substantial.
2. The Clear Connection service is completed when the connectionToBeCleared connection transits to either the Null, Failed, Held, or Queued state.
3. If the call involves only two connections or the call has parties outside the switching sub-domain, the other connection(s) may transition to Null or Failed. If the call involves more than two connections, the other connections remain unaffected.
4. The Held state is valid when a call is “suspended” for a time period prior to going to Null.
5. The Queued state is only valid for a shared bridged device configuration when there are other appearances connected into the call.

17.1.8.3.2 Device-Type Monitoring Event Sequences

Table 17-40 Clear Connection—Device-Type Monitoring Event Sequences (Generic Case)

Monitored Device	Connection	Event	Event Cause
D3 (clearingdevice)	D3C1 (connectionToBeCleared)	Failed (see item #2)	Blocked, or Normal, or Busy
		Connection Cleared	Normal Clearing
D1	D3C1 (connectionToBe-Cleared)	Failed (see item #2)	Blocked, or Normal, or Busy
		Connection Cleared	Normal Clearing
D2 and any other devices in call C1	D3C1 (connectionToBe-Cleared)	Failed (see item #2)	Blocked, or Normal, or Busy
		Connection Cleared	Normal Clearing

Table 17-41 Clear Connection—Device-Type Monitoring Event Sequences Associated with Shared Bridged Configurations [1]

Monitored Device	Connection	Event	Event Cause
D3 (clearingdevice)	D3C1 (connectionToBeCleared)	Bridged	Normal
D1	D3C1 (connectionToBe-Cleared)	Bridged	Normal
D2 and any other devices in call C1	D3C1 (connectionToBe-Cleared)	Bridged	Normal

1. This event sequence only occurs when an appearance in the device configuration remains connected in the call after the service request.

Table 17-42 Clear Connection—Device-Type Monitoring Event Sequences Associated with Suspend

Monitored Device	Connection	Event	Event Cause
D3 (clearingdevice)	D3C1 (connectionToBeCleared)	Held	Suspend
		Connection Cleared	Normal
D1	D3C1 (connectionToBe-Cleared)	Held	Suspend
		Connection Cleared	Normal
D2 and any other devices in call C1	D3C1 (connectionToBe-Cleared)	Held	Suspend
		Connection Cleared	Normal

17.1.8.3.3 Call-Type Monitoring Event Sequences

Table 17-43 Clear Connection—Call-Type Monitoring Event Sequences (Generic Case)

Monitored Call	Connection	Event	Event Cause
C1	D3C1 (callToBeCleared)	Failed (see item #2)	Blocked, or Normal, or Busy
		Connection Cleared	Normal Clearing
		Call Cleared (see item #1)	

Table 17-44 Clear Connection—Call-Type Monitoring Event Sequences Associated with Shared Bridged Configurations

Monitored Call	Connection	Event	Event Cause
C1	D3C1 (callToBeCleared)	Bridged	Normal

Note that the above sequence only occurs when an appearance in the device configuration remains connected in the call after the service request.

Table 17-45 Clear Connection—Call-Type Monitoring Event Sequences with Suspend

Monitored Call	Connection	Event	Event Cause
C1	D3C1(callToBeCleared)	Held (see item #3)	Suspend
		Connection Cleared	Normal Clearing
		Call Cleared (see item #1)	Normal

1. For call-type monitors, if the connection being cleared is the last connection in the call then the Call Cleared event will be reported.
2. If a device stays off-hook and receives busy or blocked tone, the switching function sends the Failed event (event cause of Blocked or Busy) to indicate the status of the device, followed by a Connection Cleared event. The time between the generation of the Failed event and the Connection Cleared event is device and switching function specific and the time may be substantial.
3. Prior to going to Null, a switching function may “suspend” a cleared connection (and transition the connection to the Held state) for a time period. The time between the generation of the Held event and the Connection Cleared event is device and switching function specific and the time may be substantial.

17.1.8.3.4 Functional Requirements

1. If the call associated with Clear Connection service has multiple devices that are outside the switching sub-domain, they may remain connected after this service is complete.

2. When the last connected appearance in a shared bridged device configurations is cleared using the Clear Connection service, all other device configurations appearances are also cleared from the call (i.e., Connection Cleared events).
3. The switching function should only accept the Clear Call service request and not the Clear Connection service request if clearing a connection that is involved in a conference call causes the entire conference call to be terminated. However, some switching functions accept a Clear Connection service when it results in an entire conference call to be terminated. The capability exchange services indicate if the switching functions protects or allows a conference call from being torn down via the Clear Connection service.

17.1.9 Conference Call

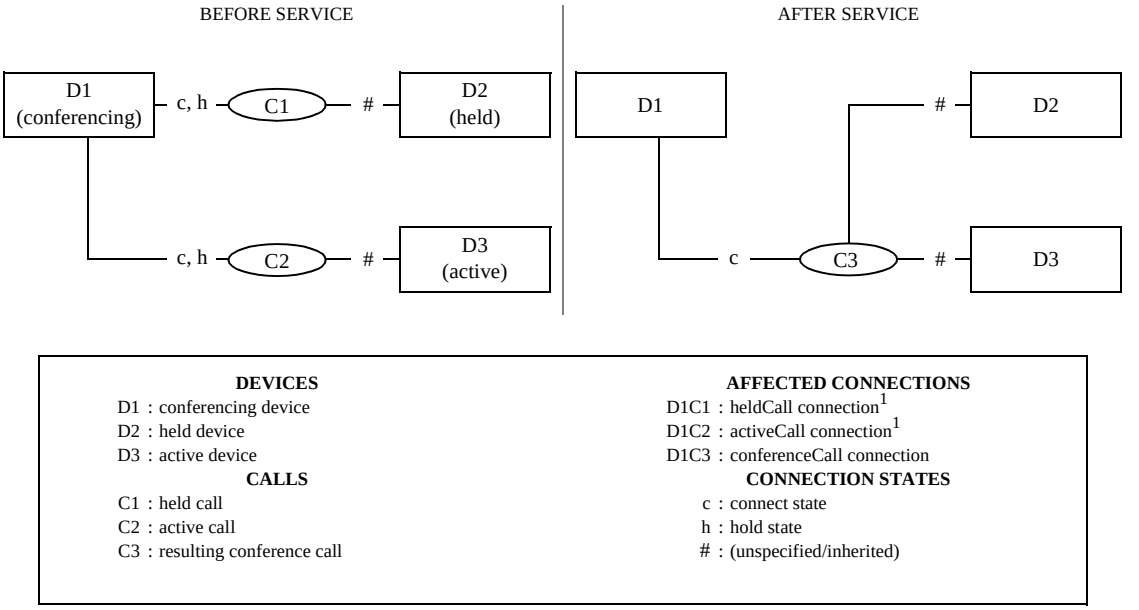
C → S

The Conference Call service provides a conference of an existing held call and another active call at a conferencing device.

The two calls are merged into a single call and the two connections at the conferencing device are resolved into a single connection. The Connection IDs formerly associated with the conferenced connections are released and a new Connection ID for the resulting connection is created.

The existing held call may consist of two or more devices.

Figure 17-9 Conference Call Service



1. See the Connection State Transitions for the appropriate initial states for these connections.

17.1.9.1 Service Request

Table 17-46 Conference Call—Service Request

Parameter Name	Type	M/ O/C	Description
heldCall	ConnectionID	M	Specifies the held connection.
activeCall	ConnectionID	M	Specifies the active connection.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

In the case where both connections are in the Held state, one connection shall be the heldCall connection and the other connection shall be the activeCall connection.

17.1.9.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.9.2.1 Positive Acknowledgement

Table 17-47 Conference Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
conferenceCall	ConnectionID	M	Specifies the resulting connection to the new call. The ConnectionID shall have the CallID of the resulting conference call and the DeviceID of the conferencing device.
connections	ConnectionList	O	Specifies information on each endpoint/ConnectionID that has changed as a result of the service.
conferenceCallInfo	ConnectionInformation	O	Specifies the connection information associated with the conferenceCall connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.9.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.9.3 Operational Model

17.1.9.3.1 Connection State Transitions

Table 17-48 Conference Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (heldCall)	Hold	Null
	Connected, only if activeCall connectionID is in Connected state.	
D1C2 (activeCall)	Connected	Null
	Hold, only if heldCall connectionID is in the Hold state. See item #2.	
D2C1 and other connections in call C1	(unspecified)	Null
D1C3	Null	Connected
D3C2	(unspecified)	Null
other connections in call C3	Null	(Inherited from the corresponding connections in C1 and C2)

1. New ConnectionIDs will be assigned to all devices that remain in the call due to the Conference. The final connection states of the new ConnectionIDs will be inherited from the states of the corresponding original calls, except for the conferencing device.
2. D1C2 can be in the Hold state, if D1C1 is also in the Hold state. This allows the conferencing of two held calls.

17.1.9.3.2 Device-Type Monitoring Event Sequences

Table 17-49 Conference Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (Conferencing device)	(see item #2)	Conferenced	Normal
D2 and any other devices in call C1 (held call)	(see item #2)	Conferenced	Normal
D3 and any other devices in call C2 (active call	(see item #2)	Conferenced	Normal

17.1.9.3.3 Call-Type Monitoring Event Sequences

Table 17-50 Conference Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	(see item #2)	Conferenced	Normal
C2	(see item #2)	Conferenced	Normal

1. For Device-Type Monitoring, a Connection Cleared event will not be seen for the activeCall or heldCall ConnectionIDs. The Conferenced event implies that the connections for these calls at this device are cleared.
2. There are multiple connections affected by this service.

17.1.9.3.4 Functional Requirements

1. The Conference service only affects the two calls specified on the service. If other calls exist at the conferencing device, they remain unaffected.
2. To get the connections for each of the devices to their initial states, the computing function can:
   - Use the Consultation Call service to place a call on hold and place a new call.
   - Use the Alternate Call service to place a call on hold and answer an alerting call.

   In either of these cases, if the switching function supports the consultOptions parameter in these services, the computing function shall provide this parameter with a value of either “Conference Only” or “Unrestricted”.

   Some switching function support a third approach for preparing for the Conference Call service involving the use of a hold service followed by a Make Call.

   The fourth approach supported by some switching functions involves two held calls at the same device.

   Certain switching functions also support a fifth approach involving two active calls at the same device.

   The computing function should use the capabilities exchange services to determine which of these approaches is supported by the switching function.
3. If the computing function uses the Consultation Call service to specify the consultOptions parameter with a value of Transfer, and then attempts to complete the consultation call with a Conference Call service, it will be rejected with a negative acknowledgement.
4. The appearances in a shared bridged device configuration are unaffected by this service.
5. The maximum number of devices in a conference is subject to switching function limits.
6. The computing function should never assume the reuse of callIDs, although some switching functions may reuse one or the other.

17.1.10 Consultation Call

C → S

The Consultation Call service places an existing active call at a device on hold and initiates a new call from the same device. The existing active call may include two or more devices.

Figure 17-10 Consultation Call Service—Case A: Complete Dialling Sequence Provided

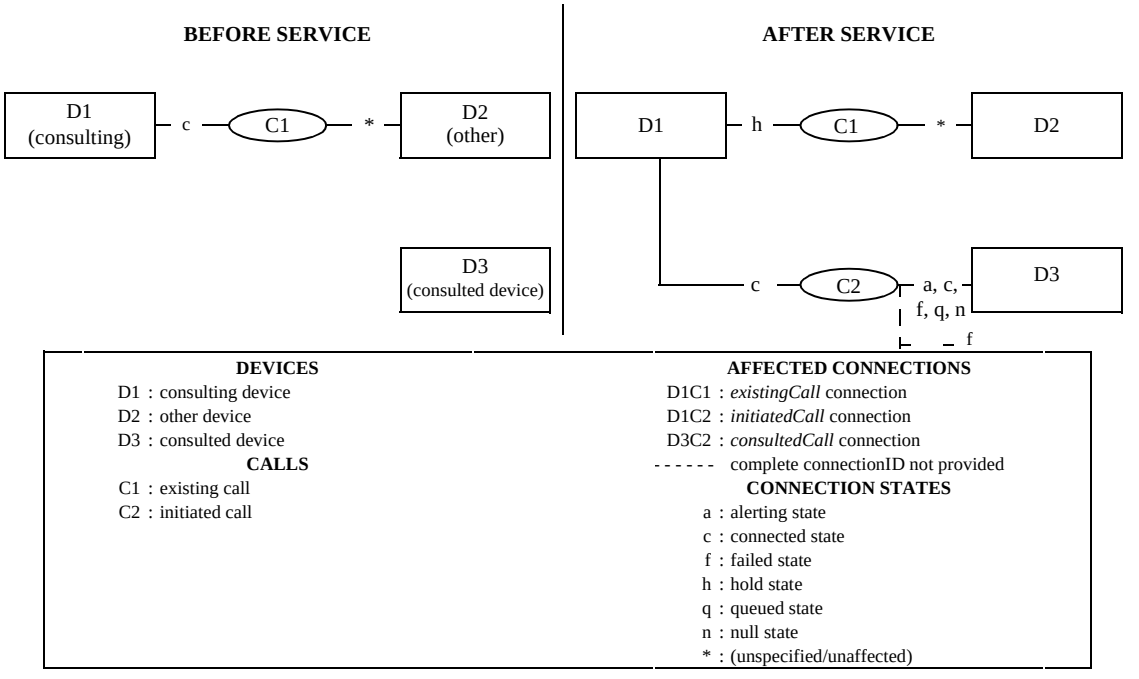
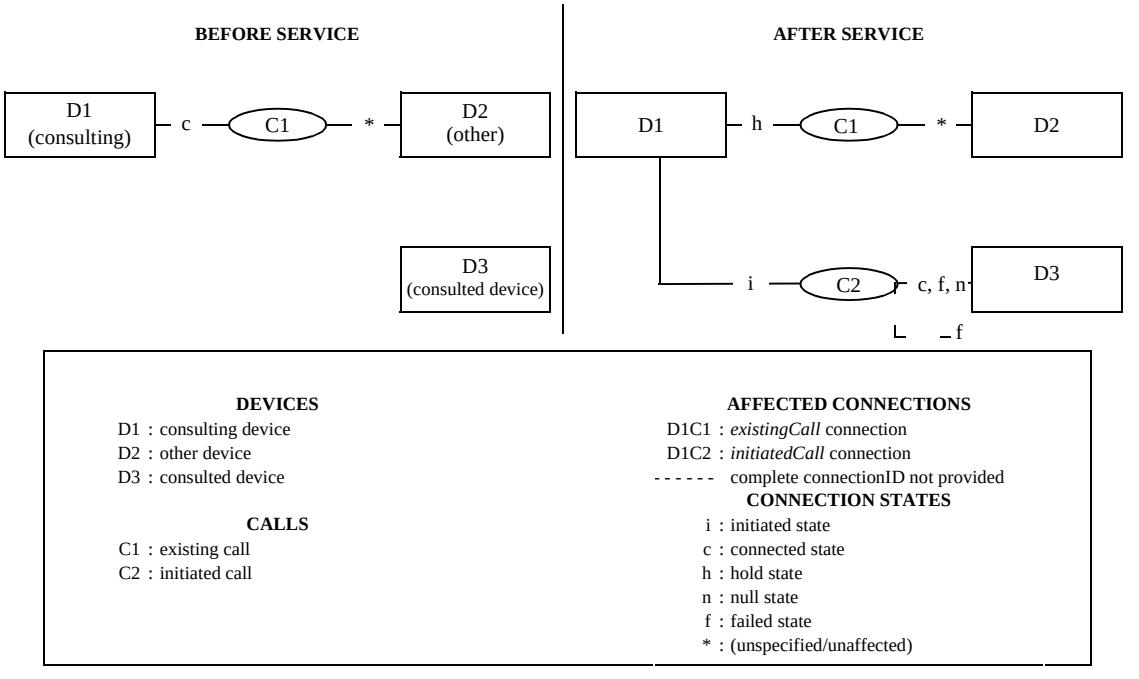


Figure 17-11 Consultation Call Service—Case B: Partial Dialling Sequence Provided



Note that connected state (D3C2) exists when D3 is a NID.

Refer to 6.8.2, “Connection Failure”, on page 53 for a complete description of partial connection IDs.

17.1.10.1 Service Request

Table 17-51 Consultation Call—Service Request

Parameter Name	Type	M/ O/C	Description
existingCall	ConnectionID	M	Specifies the active connection.
consultedDevice	DeviceID	M	Specifies the device to be consulted.
connectionReservation	Boolean	O	Specifies that the media stream channel(s) associated with the call being placed on hold be reserved for reuse at a later time. The complete set of possible values is: • True - channel(s) is to be reserved. • False - channel(s) is not to be reserved (default).
accountCode	AccountInfo	O	Specifies the account code to associate with the consulted call.
authCode	AuthCode	O	Specifies the authorization code to allow the call.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
userData	UserData	O	Specifies the user data to be sent to parties in the call.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (Priority call, for example) to be associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values. If the supported characteristics cannot be honoured, the switching function shall reject the service request.
mediaCallCharacteristics	MediaCallCharacteristics	O	This specifies the media characteristics to be associated with the call being made for the consultation. If this parameter is not present then the media class is Voice.
callingConnectionInfo	ConnectionInformation	O	This specifies the connection information needed for the creation of the new connection at the consulting device. If this parameter is not present then the connection information is switching function specific.
consultOptions	Enumerated	C	This parameter indicates the potential actions following the Consultation Call so that certain facilities can be allocated prior to a Transfer or Conference. The complete set of possible values is: • Consult Only • Transfer Only • Conference Only • Unrestricted (default) If the switching function supports this parameter, the computing function shall supply one of the values supported. If the switching function does not support this parameter, the implicit value is “Unrestricted”. This parameter does not indicate a restriction or otherwise affect the switching function’s capabilities in any way with respect to services other than Transfer or Conference.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.10.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.10.2.1 Positive Acknowledgement

Table 17-52 Consultation Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
initiatedCall	ConnectionID	M	Specifies the initial connection to the new call. The ConnectionID shall have the CallID of the resulting new call and the DeviceID of the consulting device.
mediaCallCharacteristics	MediaCallCharacteristics	C	This specifies the adjusted media characteristics for the call being made for the consultation. This parameter shall be provided if the media characteristics have been adjusted, otherwise the parameter is optional.
initiatedCallInfo	ConnectionInformation	O	This specifies the adjusted connection information used during the creation of the initiatedCall for the consulting device. If this parameter is not present then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.10.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.10.3 Operational Model

17.1.10.3.1 Connection State Transitions

Table 17-53 Consultation Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (existingCall)	Connected	Hold
D1C2 (initiatedCall)	Null	Initiated, Connected
D3C2 (consulted device)	Null	Alerting, Connected, Queued, Fail, or Null (Null if call moves away from the D3 (i.e., forwarded)).
Other connections in call C1 (existingCall) (i.e., D2C1)	(Unspecified)	(Unaffected; no transition due to this service).

1. When providing a partial dialling sequence (Case B), the connected state, for Device D3, only applies to external outbound calls. It indicates that enough of the dial string has been communicated for the switching function to connect the call to the network interface device that will be associated with the called device in the external network.

17.1.10.3.2 Device-Type Monitoring Event Sequences

There are two types of Device-Type Monitoring Event sequences depending on the dialling sequence used for the called device.

- Case A: The dialling sequence for the called device is the complete sequence for the device.

Table 17-54 Consultation Call—Device-Type Monitoring Event Sequences (Case A)

Monitored Device	Connection	Event	Event Cause
D1 (consulting device)	D1C1	Held	Normal or Consultation or Conference or Transfer [1]
	D1C2	Service Initiated (optional)	Consultation or Conference or Transfer [1]
	D1C2	Originated	Consultation or Conference or Transfer [1]
	D3C2	Events depend on the type of consulted device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). [2]	
D2 and any other devices in C1 (original call)	D1C1	Held	Normal or Consultation or Conference or Transfer [1]
D3 (consulted device)	D3C2	Events depend on the type of consulted device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). [2]	

1. The Conference event cause will only be present when the consultOptions parameter is set to “Conference Only.” The Transfer event cause will only be present when the consultOptions parameter is set to “Transfer Only.”

2. Providing of these events by the switching function does not affect the service completion conditions (i.e., a service is considered complete without these events being provided).

- Case B: The dialling sequence for the called device is a partial sequence for the device.

Table 17-55 Consultation Call—Device-Type Monitoring Event Sequences (Case B)

Monitored Device	Connection	Event	Event Cause
D1 (consulting device)	D1C1	Held	Normal or Consultation or Conference or Transfer [1]
	D1C2	Service Initiated (optional)	Consultation or Conference or Transfer [1]
	D1C2	Digits Dialled	Consultation or Conference or Transfer [1]
	D3C2	Events depend on the type of consulted device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). [2]	
D2 and any other devices in C1 (original call)	D1C1	Held	Normal or Consultation or Conference or Transfer [1]
D3 (consulted device)	D3C2	Events depend on the type of consulted device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). [2]	

1. The Conference event cause will only be present when the consultOptions parameter is set to “Conference Only.” The Transfer event cause will only be present when the consultOptions parameter is set to “Transfer Only.”

2. Providing of these events by the switching function does not affect the service completion conditions (i.e., a service is considered complete without these events being provided).

17.1.10.3.3 Call-Type Monitoring Event Sequences

There are two types of Device-Type Monitoring Event sequences depending on the dialling sequence used for the called device.

- Case A: The dialling sequence for the called device is the complete sequence for the device.

Table 17-56 Consultation Call—Call-Type Monitoring Event Sequences (Case A)

Monitored Call	Connection	Event	Event Cause
C1 (original call)	D1C1	Held	Normal or Consultation or Conference or Transfer [1]
C2 (initiated call)	D1C2	Service Initiated (optional)	Consultation or Conference or Transfer [1]
	D1C2	Originated	Consultation or Conference or Transfer [1]
	D3C2	Events depend on the type of consulted device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). [2]	

1. The Conference event cause will only be present when the consultOptions parameter is set to “Conference Only.” The Transfer event cause will only be present when the consultOptions parameter is set to “Transfer Only.”

2. Providing of these events by the switching function does not affect the service completion conditions (i.e., a service is considered complete without these events being provided).

- Case B: The dialling sequence for the called device is a partial sequence for the device.

Table 17-57 Consultation Call—Call-Type Monitoring Event Sequences (Case B)

Monitored Call	Connection	Event	Event Cause
C1 (original call)	D1C1	Held	Normal or Consultation or Conference or Transfer [1]
C2 (initiated call)	D1C2	Service Initiated (optional)	Consultation or Conference or Transfer [1]
	D1C2	Digits Dialled	Consultation or Conference or Transfer [1]
	D3C2	Events depend on the type of consulted device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). [2]	

1. The Conference event cause will only be present when the consultOptions parameter is set to “Conference Only.” The Transfer event cause will only be present when the consultOptions parameter is set to “Transfer Only.”

2. Providing of these events by the switching function does not affect the service completion conditions (i.e., a service is considered complete without these events being provided).

17.1.10.3.4 Functional Requirements

1. For the consultedDevice, all active features for this device will be honoured while the call is being made to it.
2. The consultOptions parameter indicates the potential action following the Consultation Call service so that certain facilities can be allocated if a transfer or conference is desired. If the switching function supports the consultOptions parameter, the computing function shall provide one of the supported values of this parameter obtained through the capability exchange services. If the switching function is unable to allocate the facilities for the requested Transfer or Conference, it will reject the Consultation Call service request. This parameter does not indicate a restriction or otherwise affect the switching function’s capabilities in any way with respect to services other than Transfer or Conference.
3. It is switching function specific whether a switching function may still accept a request for a Conference or Transfer if the consultOptions of Conference or Transfer was not requested as part of the Consultation Call service.
4. If the computing function specifies the consultOptions parameter with a value of Conference, then the computing function can not complete the consultation with a Transfer Call service or transfer feature.

5. If the computing function specifies the consultOptions parameter with a value of Transfer, then the computing function can not complete the consultation with a Conference Call service or conference feature.

6. The consultedDevice parameter may contain a device identifier of null or contain a Diallable Digits format (DD) device identifier that has a partial dialling sequence character (i.e., the ";" character) at the end of it. When this parameter is used in this manner, the computing function is indicating that it wishes to stage the dialling sequence. The completion of the dialling sequence can be accomplished either by entering the rest of the sequence manually at the actual device or the computing function can use the Dial Digits service to complete the sequence. As for the other types of consultedDevice parameter, they shall contain a complete dialling sequence. The switching function may have a time-out period for multi-stage dialling. If the dialling sequence does not complete prior to this timeout, it may either abort the call or attempt to use the digits already dialled and signal that dialling is complete with an originated event.

7. If the Consultation Call service is used to initiate a multistage dialling sequence, the computing function is signalled to continue the dialling sequence via either the Service Initiated event (i.e., the consultedDevice is Null) or the Digits Dialled event (i.e., the consultedDevice has a partial dialling sequence character).

8. A feature of initiating a digital data call as a result of this service (i.e., initiatedCall) is that the switching function may or may not adjust the digital data characteristics (e.g., connection rate) and connection information (e.g., number of channels) that were supplied on the Consultation Call service request (Use the capabilities exchange services to determine which feature the switching function supports). If the switching function does not support the adjusting of the characteristics/connection information, the service request will be rejected with an appropriate error code in the negative acknowledgement. If the switching function does support the adjusting of the characteristics/connection information, the positive acknowledgement will contain the adjusted value or values. If the computing function determines that the adjusted values are not adequate, it can terminate the digital data call (e.g., Clear Call).

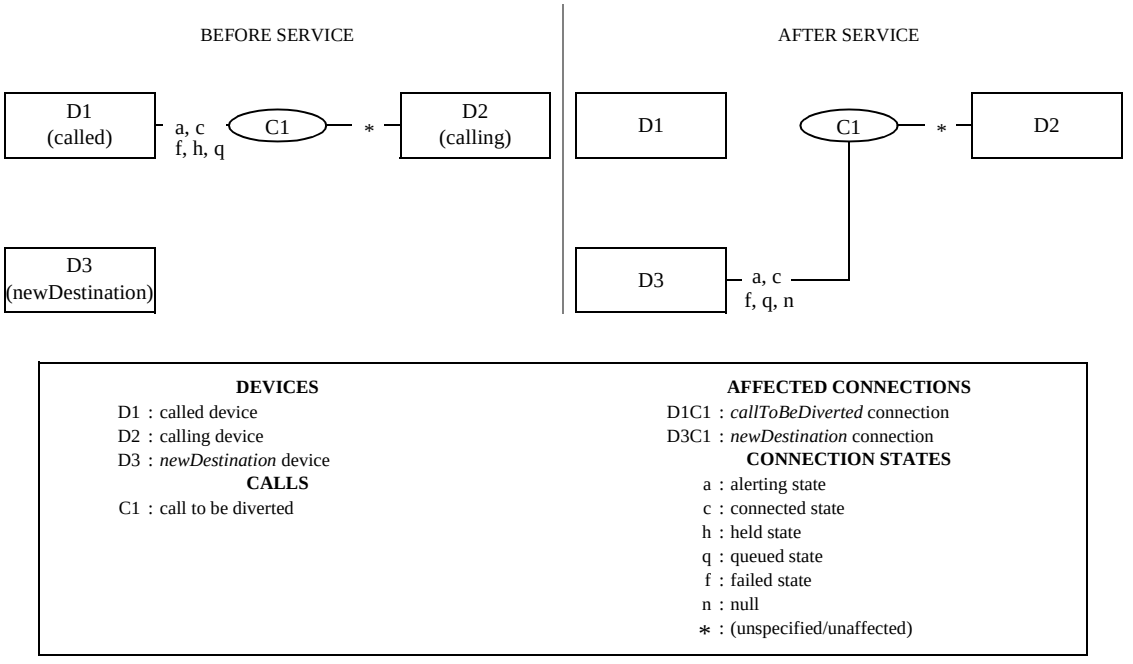
9. If the computing function makes a digital data call using this service and wants to also bind a particular Media Service to the call, then the computing function shall use the Media Attach service.

17.1.11 Deflect Call

C → S

The Deflect Call service allows the computing function to divert a call to another destination that may be inside or outside the switching sub-domain.

Figure 17-12 Deflect Call Service



Note that D1 may also be the calling device.

17.1.11.1 Service Request

Table 17-58 Deflect Call—Service Request

Parameter Name	Type	M/ O/C	Description
callToBeDiverted	ConnectionID	M	Specifies the connection to be diverted.
newDestination	DeviceID	M	Specifies the device to which the call is to be diverted (newDestination device).
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
userData	UserData	O	Specifies the user data to be sent to parties in the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.11.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.11.2.1 Positive Acknowledgement

Table 17-59 Deflect Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.11.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.11.3 Operational Model

17.1.11.3.1 Connection State Transitions

Table 17-60 Deflect Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (callToBeDiverted)	Alerting, Connected, Failed, Held, Queued	Null
D2C1 (calling device)	(Unspecified)	Unaffected; no transition due to this service.
D3C1 (newDestination)	Null	Alerting, Connected, Queued, Fail, or Null (Null if call moves away from D3 (i.e., forwarded)).

17.1.11.3.2 Device-Type Monitoring Event Sequences

Table 17-61 Deflect Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (called device)	D1C1 (callToBeDiverted)	Diverted (see item #3)	Redirected or Normal
D2 (calling device)	D1C1	Diverted (optional) (see items #2 and #3)	Redirected or Normal
	D3C1	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (see item #4)	
D3 (newDestination device)	D3C1	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (see item #4)	

17.1.11.3.3 Call-Type Monitoring Event Sequences

Table 17-62 Deflect Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D1C1	Diverted (optional) (see items #1 and #3)	Redirected or Normal
	D3C1	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (see item #4)	

1. The switching function provides the Diverted event for C1 only if it is providing Diverted events for call-type monitors. This is indicated through the capabilities exchange services.

2. The switching function provides the Diverted event for D2 only if it is providing Diverted events for all devices in the call. This is indicated through the capabilities exchange services.
3. A Connection Cleared Event will not be provided for D1 (called device). The Diverted event implies the ConnectionID has gone to Null.
4. Providing of these events by the switching function does not affect the service completion conditions (i.e., a service is considered complete without these events being provided).

17.1.11.3.4 Functional Requirements

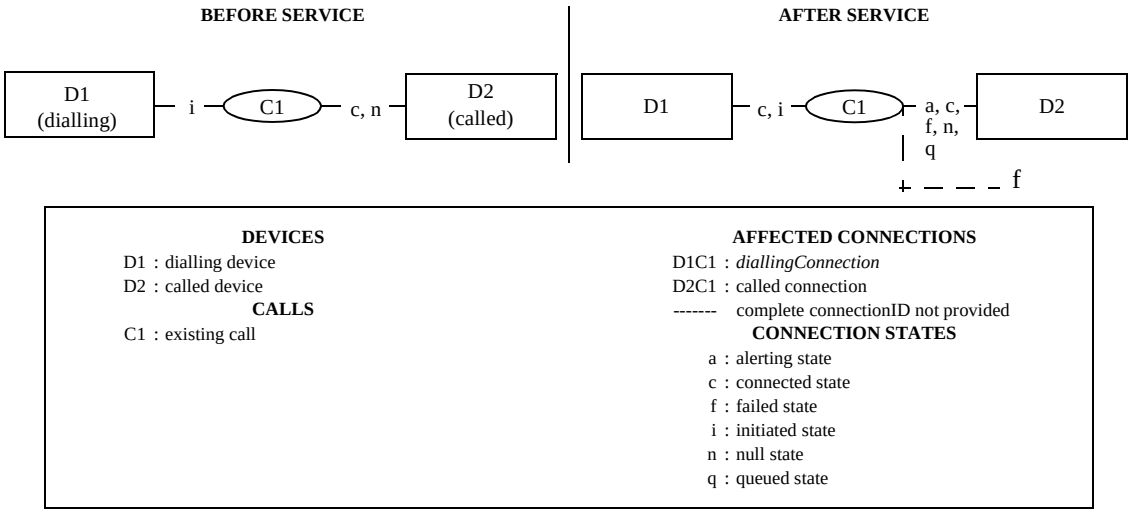
1. The Deflect Call service differs from the (Directed and Group) Pickup services in that the Pickup services redirects a call from a device and connects it to a specified device.
2. The Deflect Call service only affects the callToBeDiverted ConnectionID's call. If other calls exist at the callToBeDiverted ConnectionID's device, they remain unaffected.
3. For the newDestination, all active features for this device will be honoured while the call is being deflected to it. For example, due to active features at the called device, the call may or may not be deflected to the new destination device.
4. As a result of the Deflect Call service, the call ID associated with this call remains unchanged.
5. If the callToBeDiverted connection identifier is associated with a shared bridged device configuration appearance in the queued state (i.e., inactive mode), then the service request will be rejected with a negative acknowledgement.
6. This service request does not support the use of a null device identifier or a Diallable Digits format (DD) device identifier that has a partial dialling sequence character (i.e., the ";" character) in it for the newDestination parameter. A complete and valid device identifier shall be provided, otherwise the switching function will reject the service request with a negative acknowledgement.
7. If this service is used to deflect a digital data call, the connection for the newDestination will inherit the characteristics of the call.

17.1.12 Dial Digits

C ⟶ S

The Dial Digits service allows the computing function to perform a dialling sequence that is associated with a call that has already been initiated (i.e., has manually gone off-hook or has been initiated via a Make Call or Consultation Call service). This service is also used to perform the dialling sequences associated with completing a multi-stage dialled call.

Figure 17-13 Dial Digits Service



Note that the initial state of connected for connection D2C1 exists when D2 is a NID.

Refer to 6.8.2, “Connection Failure”, on page 53 for a complete description of partial connection IDs.

17.1.12.1 Service Request

Table 17-63 Dial Digits—Service Request

Parameter Name	Type	M/ O/C	Description
diallingConnection	ConnectionID	M	Specifies the connection which is dialling the digits.
diallingSequence	DeviceID	M	Specifies the actual string of digits to be dialled. To specify a partial dialling sequence, the Diallable Digits format (DD) of the DeviceID with the “;” character as the last digit in the string shall be used.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.12.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.12.2.1 Positive Acknowledgement

Table 17-64 Dial Digits—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.

Table 17-64 Dial Digits—Positive Acknowledgement (continued)

Parameter Name	Type	M/ O/C	Description
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.12.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.12.3 Operational Model

17.1.12.3.1 Connection State Transitions

Table 17-65 Dial Digits—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (dialling)	Initiated	Initiated, Connected
D2C1 (called)	Null, Connected (The connected state only applies to external outbound calls. It indicates that enough of the dial string has been communicated for the switching function to connect the call to the network interface device that will be associated with the called device in the external network.)	Alerting, Connected, Failed, Queued, or Null (Null if call moves away from D3 (i.e., forwarded).

17.1.12.3.2 Device-Type Monitoring Event Sequences

Table 17-66 Dial Digits—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Cause Code
D1 (dialling device)	D1C1	Digits Dialled (see item #1)	Normal or Network Dialling
	D1C1	Originated (when the dialling sequence is complete and the D1C1 connection is connected into the call (i.e., D1C1 final state = connected)).	Normal or Network Dialling
	D2C1	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (see item #2)	
D2 (called device)	D2C1	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (see item #2)	

17.1.12.3.3 Call-Type Monitoring Event Sequences

Table 17-67 Dial Digits—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Cause Code
C1	D1C1	Digits Dialled (see item #1)	Normal or Network Dialling
	D1C1	Originated (when the dialling sequence is complete and the D1C1 connection is connected into the call (i.e., D1C1 final state = connected))	Normal or Network Dialling
	D2C1	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (see item #2)	

1. This event will be provided for each diallable digits segment received by the switching function.
2. Providing of these events by the switching function does not affect the service completion conditions (i.e., a service is considered complete without these events being provided).

17.1.12.3.4 Functional Requirements

1. The Dial Digits service only effects the diallingConnection call. If other calls exist at the diallingConnection Connection ID’s device, they remain unaffected.
2. This service can not be used to generate DTMF tones on an existing connected call. The Generate Digits service is used for this. This service is only used to perform the dialling sequences for placing a call (i.e., outbound dialling).
3. If no digits have been dialled for the diallingConnection through the original Make Call service or a previous Dial Digits service, the diallingSequence parameter can be any format of device identifier. Otherwise the diallingSequence parameter shall be in Diallable Digits format.
4. If the diallingSequence parameter is intended to provide just a portion of a longer dialling sequence to follow, then it shall be in Diallable Digits format with the last character of the sequence being “;”. Otherwise the diallingSequence parameter will be interpreted as the last in the dialling sequence for the diallingConnection.
5. The premature termination of a dialling sequence can be done by either using the Clear Connection service, Clear Call service, or having the dialling connection manually cleared.
6. The switching function may have a time-out period for multi-stage dialling. If the dialling sequence does not complete prior to this timeout, it may either abort the call or attempt to use the digits already dialled and signal that dialling is complete with an originated event.
7. If the switching function determines that dialling is complete even if a “;”was supplied, it will originate the call and ignore any subsequent digits.

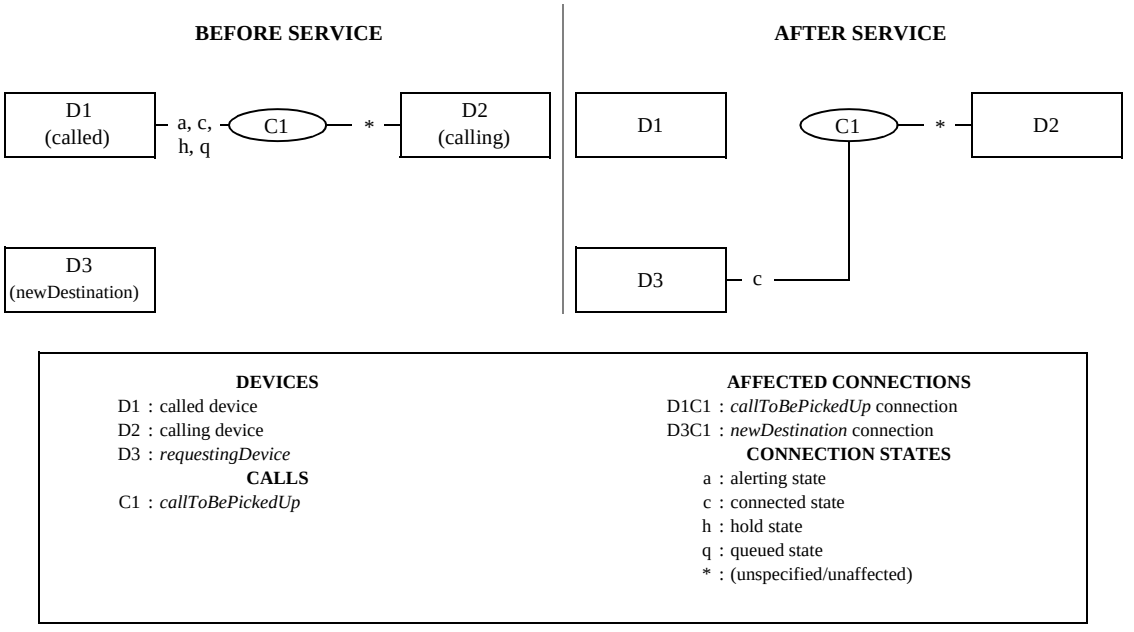
17.1.13 Directed Pickup Call

C → S

The Directed Pickup Call service moves a specified call and connects it at a new specified destination.

This results in the connection being diverted to a new destination inside the switching sub-domain.

Figure 17-14 Directed Pickup Call Service



Note that D1 could also be the calling device.

17.1.13.1 Service Request

Table 17-68 Directed Pickup Call—Service Request

Parameter Name	Type	M/ O/C	Description
callToBePickedUp	ConnectionID	M	Specifies the connection to be picked up.
requestingDevice	DeviceID	M	Specifies the device which is picking up the call.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
userData	UserData	O	Specifies the user data to be sent to parties in the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.13.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.13.2.1 Positive Acknowledgement

Table 17-69 Directed Pickup Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
pickedCall	ConnectionID	O	Specifies the connectionID for the device which is picking up the call.

Table 17-69 Directed Pickup Call—Positive Acknowledgement (continued)

Parameter Name	Type	M/ O/C	Description
pickedCallInfo	ConnectionInformation	O	Specifies the connection information associated with the pickedCall connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.13.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.13.3 Operational Model

17.1.13.3.1 Connection State Transitions

Table 17-70 Directed Pickup Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (callToBePickedUp)	Alerting, Connect, Hold, or Queued	Null
D2C1 (calling device)	(Unspecified)	(Unaffected; no transition due to this service.)
D3C1 (requestingDevice)	Null	Connected

17.1.13.3.2 Device-Type Monitoring Event Sequences

Table 17-71 Directed Pickup Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (called device)	D1C1 (callToBepickedUp)	Diverted (see item #2)	Call Pickup
D2 (calling device)	D1C1	Diverted (optional) (see items #1 and #2)	Call Pickup
	D3C1 (see item #5)	Service Initiated (optional) (see item #5)	Call Pickup (prompting)
	D3C1	Established	Call Pickup
D3 (requestingDevice)	D3C1	Service Initiated (optional) (see item #5)	Call Pickup (prompting)
	D3C1	Established	Call Pickup

17.1.13.3.3 Call-Type Monitoring Event Sequences

Table 17-72 Directed Pickup Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D1C1	Diverted (optional) (see Items #2 and #4)	Call Pickup
	D3C1 (item #5)	Service Initiated (optional) (item #5)	Call Pickup (prompting)
	D3C1	Established	Call Pickup

1. The switching function provides the Diverted event for D2 only if it is providing Diverted events for all devices in the call. This is indicated through the capabilities exchange services.
2. A Connection Cleared event will not be provided. The Diverted event implies the ConnectionID (for D1C1) has gone to Null.

3. If there is already a connection in the initiated state at the requestingDevice device, the computing function may receive a Connection Cleared event for that connection and an Established event for the connection involving the picked call.
4. The switching function provides the Diverted event for C1 only if it is providing Diverted events for call-type monitors. This is indicated through the capabilities exchange services.
5. The Service Initiated event is dependent upon the prompting mode (as described in 6.8.10, “Prompting”, on page 65).
    - For the “prompting is a pre-condition of the service” mode, the Service Initiated is generated before any service specific events and is not part of the service completion criteria. The connectionID associated with the Service Initiated event is not associated with the Directed Pickup service.
    - For the “prompting is part of the service” mode”, the Service Initiated event is part of the service completion criteria, and is generated after the Diverted event and contains the same connectionID as the Diverted and Established events.
6. If a Bridged event is generated for an Independent Shared Bridged device configuration (see Functional Requirement #4), it is not part of the service completion criteria.

17.1.13.3.4 Functional Requirements

1. This service differs from the Deflect Call service in that the Deflect Call service redirects a call to another destination (in which the resulting state of the new destination depends on the destination’s type and active features). The Directed Pickup Call service redirects a call to another destination which is immediately connected to the call.
2. For the newDestination, all features such as Forwarding and Do Not Disturb for this device will be ignored while the call is being redirected to it.
3. As a result of the Directed Pickup service, the call ID associated with this call remains unchanged.
4. For Shared Bridged device configurations, when a call is picked from an appearance (callToBePickedUp) all appearances will be cleared from the call (i.e., Connection Cleared events with a cause of Normal), except when the call that is being picked is part of an Independent Shared Bridged device configuration and the appearance from which the call is being picked is not the last appearance connected into the call. In this case the appearance from which the call is being picked will return to the inactive mode (i.e., Bridged event with a cause of Normal) and the other appearances are unaffected. For more information, refer to Annex A.2.3, “Shared-Bridged”.
5. This service request does *not* support the use of a null device identifier or a Diallable Digits format (DD) device identifier that has a partial dialling sequence character (i.e., the “;” character) in it for the newDestination parameter. A complete and valid device identifier shall be provided, otherwise the switching function will reject the service request with a negative acknowledgement.
6. This service is used when the device associated with the callToBePickedUp connection ID is different from the newDestination device. If the devices are the same, then the service is rejected. The Answer Call and the Retrieve Call services should be used instead.

17.1.14 Group Pickup Call

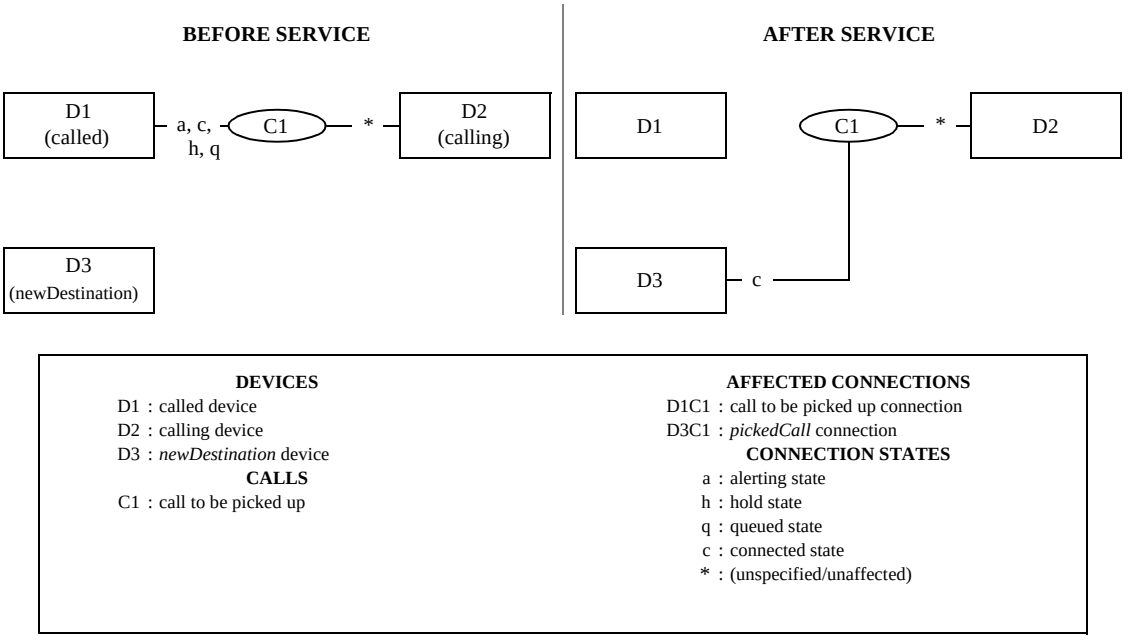
C → S

The Group Pickup Call service moves a call that is a member of a specified or default pickup group to a new specified destination.

This results in a connection in a pickup group to be connected to a new specified destination inside the switching sub-domain.

Note that the difference between this service and the Directed Pickup Call service is that Directed Pickup Call service specifies the actual connection to be picked up whereby the Group Pickup Call service does not.

Figure 17-15 Group Pickup Call Service



Note that D1 may also be the calling device.

17.1.14.1 Service Request

Table 17-73 Group Pickup Call—Service Request

Parameter Name	Type	M/ O/C	Description
newDestination	DeviceID	M	Specifies the device which is picking up the call.
pickGroup	DeviceID	O	Specifies the pick group. If this parameter is not provided, the switching function may use a pick group associated with the newDestination device.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
userData	UserData	O	Specifies the user data to be sent to parties in the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.14.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.14.2.1 Positive Acknowledgement

Table 17-74 Group Pickup Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
pickedCall	ConnectionID	O	Specifies the connection ID of call C1 at device D3.
pickedCallInfo	ConnectionInformation	O	Specifies the connection information associated with the pickedCall connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.14.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.14.3 Operational Model

17.1.14.3.1 Connection State Transitions

Table 17-75 Group Pickup Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (call to be picked up)	Alerting, Connected, Hold, or Queued	Null
D2C1 (calling device)	(Unspecified)	(Unaffected; no transition due to this service.)
D3C1 (pickedCall)	Null	Connected

17.1.14.3.2 Device-Type Monitoring Event Sequences

Table 17-76 Group Pickup Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (called device)	D1C1 (callToBepickedUp)	Diverted (see item #1)	Call Pickup
D2 (calling device)	D1C1	Diverted (optional) (see item #1 and #2)	Call Pickup
	D3C1 (see item 2)	Service Initiated (optional) (see item #5)	Call Pickup (prompting)
	D3C1	Established	Call Pickup
D3 (newDestination device)	D3C1	Service Initiated (optional)	Call Pickup (prompting)
	D3C1	Established	Call Pickup

17.1.14.3.3 Call-Type Monitoring Event Sequences

Table 17-77 Group Pickup Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D1C1	Diverted (optional) (see item #1 and #3)	Call Pickup
	D3C1 (see item 2	Service Initiated (optional) (see item 5)	Call Pickup (prompting)
	D3C1	Established	Call Pickup

1. A Connection Cleared event will not be provided. The Diverted event implies the ConnectionID (for D1C1) has gone to Null.

2. The switching function provides the Diverted event for D2 only if it is providing Diverted events for all devices in the call. This is indicated through the capabilities exchange services.
3. The switching function provides the Diverted event for C1 only if it is providing Diverted events for call-type monitors. This is indicated through the capabilities exchange services.
4. If there is already a connection in the initiated state at the newDestination device, the computing function may receive a Connection Cleared event for that connection and an Established event for the connection involving the picked call.
5. The Service Initiated event is dependent upon the prompting mode (as described in 6.8.10, “Prompting”, on page 65).
   - For the “prompting is a pre-condition of the service” mode, the Service Initiated is generated before any service specific events and is not part of the service completion criteria. The connectionID associated with the Service Initiated event is not associated with the Group Pickup service.
   - For the “prompting is part of the service” mode”, the Service Initiated event is part of the service completion criteria, and is generated after the Diverted event and contains the same connectionID as the Diverted and Established events.
6. If a Bridged event is generated for an Independent Shared Bridged device configuration (see Functional Requirement #6), it is not part of the service completion criteria.

17.1.14.3.4 Functional Requirements

1. This service differs from the Deflect Call service in that the Deflect Call service redirects a call to another destination (in which the resulting state of the new destination depends on the destination’s type and active features). The Group Pickup Call service redirects a call to another destination which is immediately connected to the call.
2. The Group Pickup service is administered by the switching function. The switching function determines which call the newDestination connects to by first determining which group the newDestination belongs to (either as specified by the pickGroup parameter or by a switching function administered group associated with the newDestination device) and then connecting it to the appropriate call.
3. For the newDestination, all features such as Forwarding and Do Not Disturb for this device will be ignored while the call is being redirected to it.
4. As a result of the Group Pickup service, the call ID associated with this call remains unchanged.
5. This service request does *not* support the use of a null device identifier or a Diallable Digits format (DD) device identifier that has a partial dialling sequence character (i.e., the “;” character) in it for the newDestination or pickGroup parameters. A complete and valid device identifier shall be provided, otherwise the switching function will reject the service request with a negative acknowledgement.
6. For Shared Bridged device configurations, when a call is picked from an appearance all appearances will be cleared from the call (i.e., Connection Cleared events with a cause of Normal), except when the call that is being picked is part of an Independent Shared Bridged device configuration and the appearance from which the call is being picked is not the last appearance connected into the call. In this case the appearance from which the call is being picked will return to the inactive mode (i.e., Bridged event with a cause of Normal) and the other appearances are unaffected. For more information, refer to Annex A.2.3, “Shared-Bridged”.

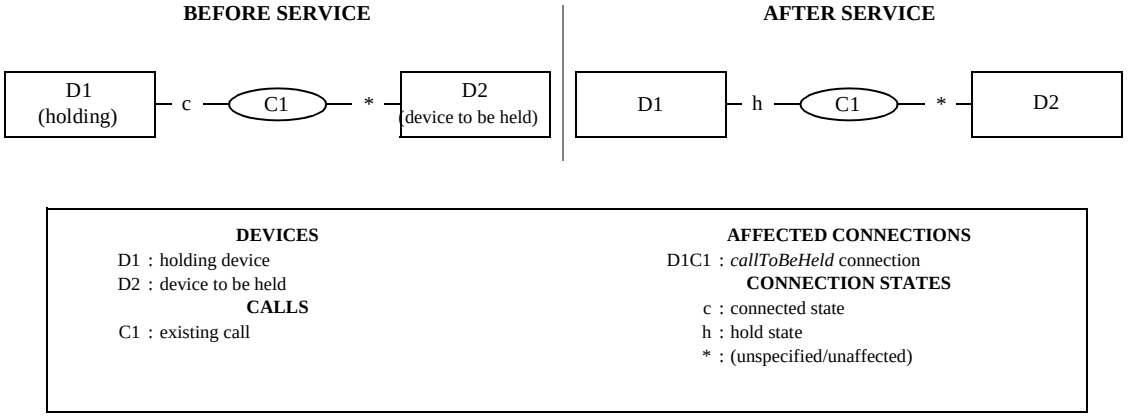
17.1.15 Hold Call

C → S

The Hold Call service places a connected connection on hold at the same device.

This service interrupts communication for an existing call at a device.

Figure 17-16 Hold Call Service



17.1.15.1 Service Request

Table 17-78 Hold Call—Service Request

Parameter Name	Type	M/ O/C	Description
callToBeHeld	ConnectionID	M	Specifies the active connection to be held.
connectionReservation	Boolean	O	Specifies that the media stream channel(s) associated with the call being placed on hold be reserved for reuse at a later time. The complete set of possible values is: • True - channel(s) is to be reserved. • False - channel(s) is not to be reserved (default).
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information.

17.1.15.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.15.2.1 Positive Acknowledgement

Table 17-79 Hold Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information.

17.1.15.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.15.3 Operational Model

17.1.15.3.1 Connection State Transitions

Table 17-80 Hold Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (callToBeHeld)	Connected	Hold
other connections in the call (i.e., D2C1)	(Unspecified)	(Unaffected; no transition due to this service).

17.1.15.3.2 Device-Type Monitoring Event Sequences

Table 17-81 Hold Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (Holding device)	D1C1 (callToBeHeld)	Held	Normal
D2 (device to be held and any other connections in call C1)	D1C1 (callToBeHeld)	Held	Normal

17.1.15.3.3 Call-Type Monitoring Event Sequences

Table 17-82 Hold Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D1C1(callToBe-Held)	Held	Normal

1. If a Bridged event is generated for an Independent Shared Bridged device configuration (see Functional Requirement #2), it is not part of the service completion criteria.

17.1.15.3.4 Functional Requirements

1. The switching function may have a time-out period for the call once it is held. If the call is not retrieved before the time-out period ends, then the held call may ringback either the holding device or a device predefined by the switching function (such as an attendant console, for example).
2. For Shared Bridged device configurations, when a call is held by an appearance (callToBeHeld) all appearances will transition to the hold state (i.e., Held events with a cause of Normal), except when the call that is being held is part of an Independent Shared Bridged device configuration and the appearance that is holding the call is not the last appearance connected into the call. In this case only the appearance holding the call will transition to the hold state and the other appearances are unaffected. For more information, refer to Annex A.2.3, “Shared-Bridged”.
3. As a result of placing a connection associated with a digital data call (if supported by the switching function) on hold, the other calls at the device or device configuration are unrelated with this call.

17.1.16 Intrude Call

C → S

The Intrude Call service adds the calling device to a call at a busy called device. Depending upon the switching function, the result will be that the calling device is either actively or silently participating in the called device's existing call or consulting with the called device with a new call

There are two cases specified for the Intrude Call service:

- Case A: In this case, the calling device (D1) joins call C2 with devices D2 and D3.
- Case B: In this case, the called device (D2) places its existing active call on hold first and then connects to the new calling device (D1).

Figure 17-17 Intrude Call Service (Case A)

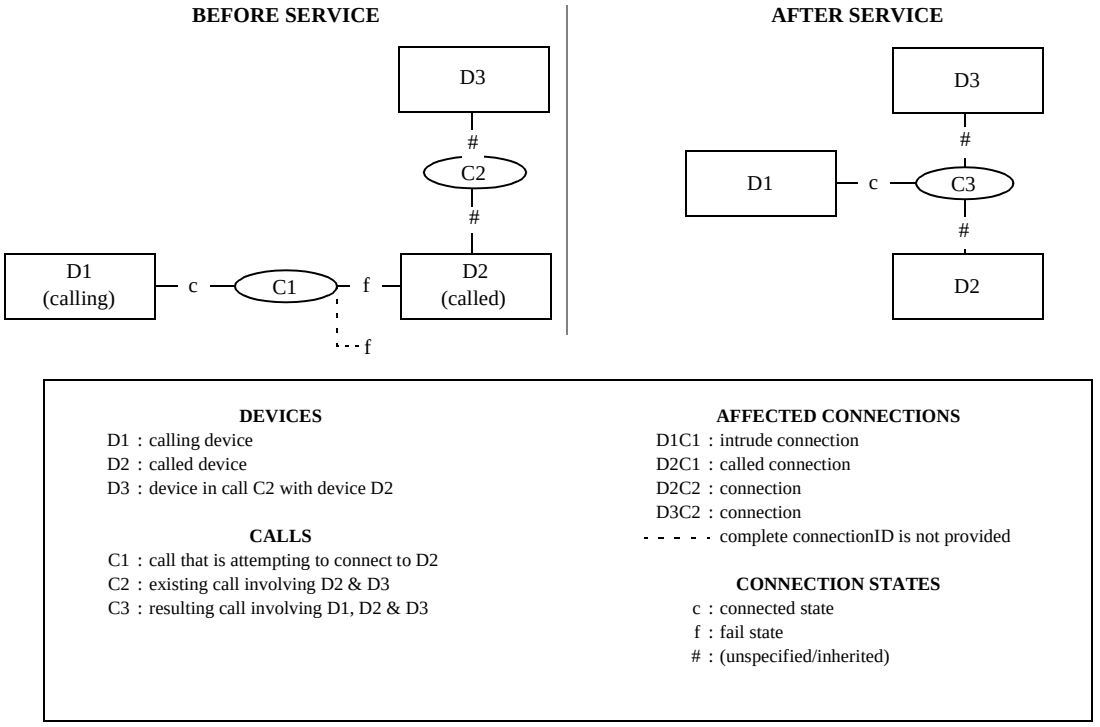
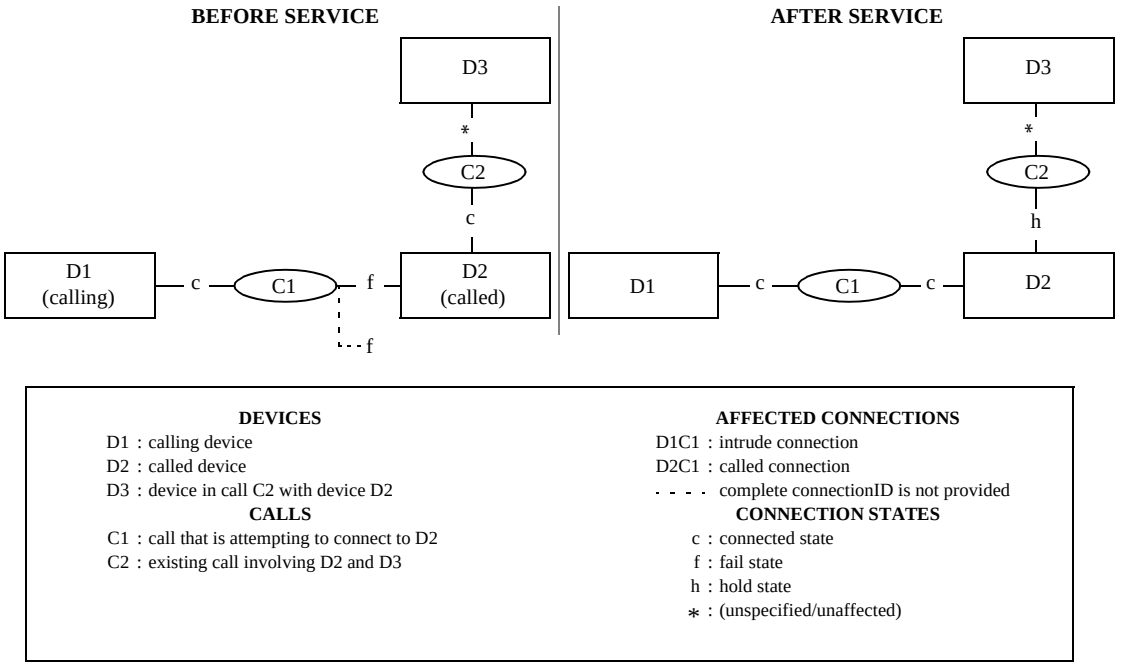


Figure 17-18 Intrude Call Service (Case B)



Refer to section 6.8.2, “Connection Failure”, on page 53 for a complete description of partial connection IDs.

17.1.16.1 Service Request

Table 17-83 Intrude Call—Service Request

Parameter Name	Type	M/ O/C	Description
intrude	ConnectionID	M	Specifies the connection of the calling device.
participationType	Enumerated	O	Specifies the type of participation the added device has in the resulting call. The complete set of possible values is: • active (default) - The added device actively participates in the resulting conference call. • silent - The added device can listen but cannot actively participate in the resulting conference call. If the switching function only supports “Case B” of the service and the participationType parameter is supplied, then the service will be rejected by the switching function.
userData	UserData	O	Specifies the user data to be sent to parties in the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.16.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.16.2.1 Positive Acknowledgement

Table 17-84 Intrude Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
conferencedCall	ConnectionID	C	ConnectionID includes the CallID of the resulting call and the DeviceID of the calling device. Mandatory, if the Intrude Call service creates a new ConnectionID for the calling device as in Case A; otherwise, the parameter is not provided.
conferencedCallInfo	ConnectionInformation	O	Specifies the connection information associated with the conferencedCall connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information.

17.1.16.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.16.3 Operational Model

17.1.16.3.1 Connection State Transitions

Table 17-85 Intrude Call—Connection State Transitions

Connection	Initial State (Required)	Final State	
		Case A	Case B
D2C1 (called device)	Fail	Null	Connected
D1C1 (intrude)	Connected	Null	Connected
D1C3	Null	Connected	N/A
D2C2	Case A: (Unspecified) Case B: Connected	Null	Hold
D3C2	(Unspecified)	Null	(Unaffected)
D2C3 and all connections in call C3 that had corresponding connections in call C2.	Null	(Inherited from corresponding connections in call C2)	N/A

17.1.16.3.2 Device-Type Monitoring Event Sequences

- Case A: D1, the calling device, joins call C3 with devices D2 and D3.

Table 17-86 Intrude Call—Device-Type Monitoring Event Sequences (Case A)

Monitored Device	Connection	Event	Event Cause
D1 (calling device)	D1C3	Conferenced	Override, Silent Participation, or Active Participation
D2 (called device)	D2C3	Conferenced	Override, Silent Participation, or Active Participation
other devices in original call C2 (i.e., D3)	D3C3	Conferenced	Override, Silent Participation, or Active Participation

- Case B: The called device places its existing active call on hold first and then connects to the new calling device (D1).

Table 17-87 Intrude Call—Device-Type Monitoring Event Sequences (Case B)

Monitored Device	Connection	Event	Event Cause
D1 (calling device)	D2C1 (called)	Established	Intrude
D2 (called device)	D2C2	Held	Normal or Intrude
	D2C1 (called)	Established	Intrude
Other devices in original call C2 (i.e., D3)	D2C2	Held	Normal or Intrude

17.1.16.3.3 Call-Type Monitoring Event Sequences

- Case A: D1, the calling device, joins call C3 with devices D2 and D3.

Table 17-88 Intrude Call—Call-Type Monitoring Event Sequences (Case A)

Monitored Call	Connection	Event	Event Cause
C1 (call attempting to connect to device D2)	D1C3	Conferenced	Override, Silent Participation, or Active Participation
C2 (resulting conference call)	D1C3	Conferenced	Override, Silent Participation, or Active Participation

- Case B: The called device places its existing active call on hold first and then connects to the new calling device (D1).

Table 17-89 Intrude Call—Call-Type Monitoring Event Sequences (Case B)

Monitored Call	Connection	Event	Event Cause
C2 (original call)	D2C2	Held	Intrude or Normal
C1 (call attempting to connect to device D2)	D2C1 (called)	Established	Intrude

1. The event cause is based on the value of the participationType parameter:
   - If the parameter was set to active, then the event cause will be Active Participation.
   - If the parameter was set to silent, then the event cause will be Silent Participation.
   - If the switching function cannot determine the type used, then the event cause will be Override.

17.1.16.3.4 Functional Requirements

1. To cancel an Intrude Call service, the computing function can either:
   - Issue the Clear Connection or Clear Call service with the ConnectionID of either the calling (cases A or B) or called (case B) device.
   - Have the calling device go on-hook.

   If the above is successfully executed, normal call progress messages of Connection Cleared (with an event cause of Normal Clearing) will be generated.
2. The computing function should never assume the reuse of callIDs, although some switching functions may reuse one or the other.
3. If the called device has more than one call during the execution of this service, the switching function will choose which call to intrude upon.

17.1.17 Join Call

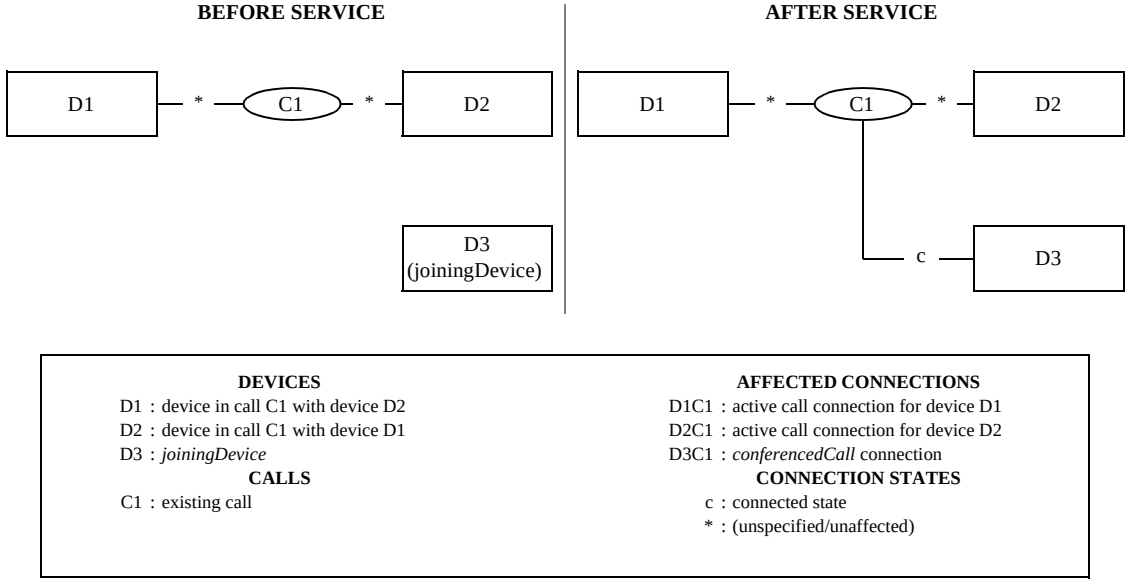
C → S

The Join Call service allows a computing function to request, on behalf of a device, that the device be joined into an existing call. In the process of establishing a connection with the joiningDevice, the joiningDevice may be prompted to go off-hook (if necessary) and when that device does so, it is added into the call.

This service is different from the Single Step Conference service in that the request is made on behalf of the joiningDevice (originating device).

This service is different from the Intrude Call service in that there is no prior failed call at the intruded-upon connection.

Figure 17-19 Join Call Service



17.1.17.1 Service Request

Table 17-90 Join Call—Service Request

Parameter Name	Type	M/ O/C	Description
activeCall	connectionID	M	Specifies an existing connection in an active call to which the new device is to be added (or joined)
joiningDevice	DeviceID	M	Specifies the device that is to be added to (join) the existing call
autoOriginate	Enumerated	O	Specifies if the joining device is to be prompted or not (hands-free mode). The complete set of possible values is: • Prompt (default) • Do Not Prompt (auto originate)
participationType	Enumerated	O	Specifies the type of participation the joining device has in the resulting call. This is one of the following: • active (default) - the joining device actively participates in the resulting conference call. As a result, the flow direction of the joiningDevice's connection (i.e., conferencedCall) will be Transmit and Receive. • silent - the joining device can listen but cannot actively participate in the resulting conference call. As a result, the flow direction of the joiningDevice's connection (i.e., conferencedCall) will be Receive.

Table 17-90 Join Call—Service Request (continued)

Parameter Name	Type	M/ O/C	Description
accountCode	AccountInfo	O	Specifies the account code to associate with the call.
authCode	AuthCode	O	Specifies the authorization code to allow the call.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
userData	UserData	O	Specifies the user data to be sent to parties in the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.17.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.17.2.1 Positive Acknowledgement

Table 17-91 Join Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
conferencedCall	ConnectionID	M	Specifies the callID of the existing active call and the DeviceID of the joining device.
conferencedCallInfo	ConnectionInformation	O	Specifies the connection information associated with the conferencedCall connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.17.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.17.3 Operational Model

17.1.17.3.1 Connection State Transitions

Table 17-92 Join Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1	(Unspecified)	(Unaffected; no transition due to this service)
D2C1	(Unspecified)	(Unaffected; no transition due to this service)
D3C1 (conferencedCall)	Null	Connected

17.1.17.3.2 Device-Type Monitoring Event Sequences

Table 17-93 Join Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1	D3C1 (conferencedCall)	Conferenced	Join Call, ActiveParticipation, Silent Participation
	D3C1 (conferencedCall) (see item 2)	Service Initiated (optional) (see item 2)	Join Call (prompting)
	D3C1 (conferenced Call)	Established	Join Call, ActiveParticipation, Silent Participation
D2	D3C1 (conferencedCall)	Conferenced	Join Call, ActiveParticipation, Silent Participation
	D3C1 (conferencedCall) (see item 2)	Service Initiated (optional) (see item 2)	Join Call (prompting)
	D3C1 (conferencedCall)	Established	Join Call, ActiveParticipation, Silent Participation
D3 (joiningDevice)	D3C1 (conferencedCall)	Conferenced	Join Call, ActiveParticipation, Silent Participation
	D3C1 (conferencedCall) (see item 2)	Service Initiated (optional) (see item 2)	Join Call (prompting)
	D3C1 (conferencedCall)	Established	Join Call, ActiveParticipation, Silent Participation

17.1.17.3.3 Call-Type Monitoring Event Sequences

Table 17-94 Join Call—Call-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
C1	D3C1	Conferenced	Join Call, ActiveParticipation, Silent Participation
	D3C1 (see item 2)	Service Initiated (optional) (see item 2)	Join Call (prompting)
	D3C1	Established	Normal

1. For the Conferenced event, the event cause is based upon the value of the participationType parameter:
   - If it was set to active, then the event cause is Active Participation.
   - If it was set to silent, then the event cause is Silent Participation.
   - If the switching function cannot determine the type used, then the event cause is Join Call.
2. The Service Initiated event is dependent upon the prompting mode (as described in 6.8.10, “Prompting”, on page 65).
   - For the “prompting is a pre-condition of the service” mode, the Service Initiated is generated before any service specific events and is not part of the service completion criteria. The connectionID associated with the Service Initiated event is not associated with the Join Call service.
   - For the “prompting is part of the service” mode”, the Service Initiated event is part of the service completion criteria, and is generated after the Conferenced event and contains the same connectionID as the Conferenced and Established events.

17.1.17.3.4 Functional Requirements

1. The appearances in a shared bridged device configuration are unaffected by this service.
2. This service request does not support the use of a null device identifier or a Diallable Digits format (DD) device identifier that has a partial dialling sequence character (i.e., the “;” character) in it for the

joiningDevice parameter. A complete and valid device identifier shall be provided, otherwise the switching function will reject the service request with a negative acknowledgement.

3. Prompting of the joining device during the processing of the Join Call service is switching function specific (display flashing, ring pattern, lamp blinking, etc.).
4. If the autoOriginate parameter has a value of “Do Not Prompt”, then the device should be a speakerphone, hands-free telephone, or other device that can be automatically answered. If the device is not of this type, then the processing of this parameter is switching function specific. For example, the switching function may choose to accept this service and ignore this parameter, or it may reject the Join Call service with negative acknowledgement.
5. Call Forwarding and Do Not Disturb features for the joining device are not honoured. If a switching function supports prompting of the joining device and detects that a feature was activated while processing the Join Call service and this feature cannot be overridden, then the switching function will return a negative acknowledgement.
6. The call ID in the ConnectionIDs for the resulting conference call is inherited from the original active call.
7. When prompting a device which has a call appearance, bridged, or hybrid device configuration, only one of the appearances will be delivered the call.
8. If this service is used to join a device into a digital data call, the connection for the joiningDevice (i.e., conferencedCall) will inherit the characteristics of the call with the exception of connection flow direction and the number of channels used. In these cases, the connection flow direction is dependent on the value of participationType and the number of channels is based on what the switching function will use to allow the joiningDevice to effectively participate in the call.

17.1.18 Make Call

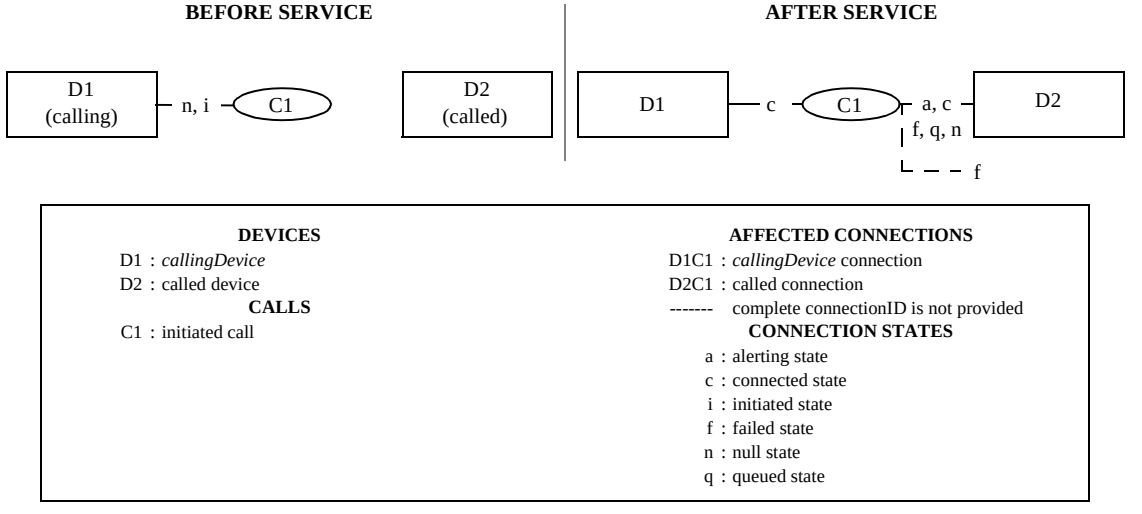
C → S

The Make Call service allows the computing function to set up a call between a calling device and a called device.

The service creates a new call and establishes an initiated or connected connection with the calling device. The Make Call service assigns a ConnectionID to the calling device and returns it in the positive acknowledgement.

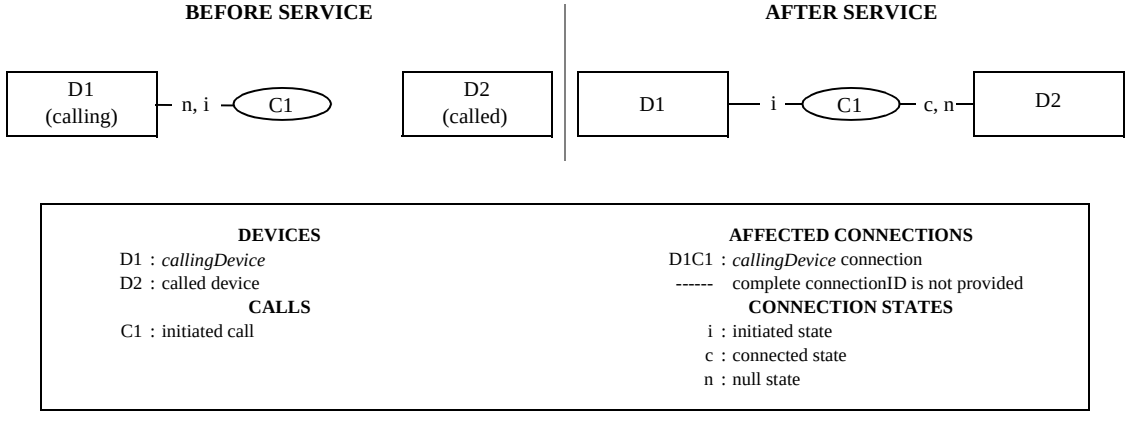
In the process of establishing the connection with the calling device, the calling device may be prompted to go off-hook (if necessary) and when that device does so, a call to the called device is originated or the calling device is still in the process of dialling the called device.

Figure 17-20 Make Call Service—Case A: Complete Dialling Sequence Provided



Note that (Case A) the Initiated state (D1C1) is used for Offhook Dialling (see Functional Requirement #6).

Figure 17-21 Make Call Service—Case B: Partial Dialling Sequence Provided



Refer to 6.8.2, “Connection Failure”, on page 53 for a complete description of partial connection IDs.

Note that (Case B) the connected state (D2C1) exists when D2 is a NID.

17.1.18.1 Service Request

Table 17-95 Make Call—Service Request

Parameter Name	Type	M/ O/C	Description
callingDevice	DeviceID	M	Specifies the calling/originating device. Note that this device may be a device that represents a group of devices. In this case the callingDevice in the service request is different from the actual calling device.
calledDirectoryNumber	DeviceID	M	Specifies the called device.
accountCode	AccountInfo	O	Specifies the account code to associate with the new call.
authCode	AuthCode	O	Specifies the authorization code to allow the call.
autoOriginate	Enumerated	O	Specifies if the calling device's connection is automatically answered (hands-free mode). The complete set of possible values is: • Prompt (default) • Do Not Prompt (auto originate)
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
userData	UserData	O	Specifies the user data to be sent to parties in the call.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (Priority call, for example) to be associated with the call. See 12.2.4, "CallCharacteristics", on page 84 for the complete set of possible values. If the supported characteristics cannot be honoured, the switching function shall reject the service request.
mediaCallCharacteristics	MediaCallCharacteristics	O	This specifies the media characteristics to be associated with the call being made. If this parameter is not present then the media class is Voice.
callingConnectionInfo	ConnectionInformation	O	This specifies the connection information needed for the creation of a connection at the callingDevice. If this parameter is not present then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information.

17.1.18.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.18.2.1 Positive Acknowledgement

Table 17-96 Make Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
callingDevice	ConnectionID	M	Specifies the initial connection to the new call. The ConnectionID shall have the CallID of the resulting new call and the DeviceID of the calling device. Note that the calling device parameter in the service request may be different from the deviceID in the connectionID of callingDevice in the positive acknowledgement (when the callingDevice in the service request represents a group of stations, for example).
mediaCallCharacteristics	MediaCallCharacteristics	C	This specifies the adjusted media characteristics for the call being made. This parameter shall be provided if the call characteristics have been adjusted, otherwise it is optional.
initiatedCallInfo	ConnectionInformation	O	This specifies the adjusted connection information used during the creation of the initiatedCall for the callingDevice. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.18.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.18.3 Operational Model

17.1.18.3.1 Connection State Transitions

Table 17-97 Make Call—Connection State Transitions (Case A: Complete Dialling Sequence)

Connection	Initial State (Required)	Final State
D1C1 (callingDevice)	Null, Initiated (see item #17.1.18.3.1)	Initiated, Connected
D2C1 (called)	Null	Alerting, Connected, Queued, Fail, or Null (Null if call moves away from the originally called device [i.e., forwarded]).

Table 17-98 Make Call—Connection State Transitions (Case B: Partial Dialling Sequence)

Connection	Initial State (Required)	Final State
D1C1 (callingDevice)	Null, Initiated (see item #17.1.18.3.1)	Initiated
D2C1 (called)	Null	Connected (see item #1), Null

1. When providing a partial dialling sequence (Case B) the connected state for D2C1 only applies to external outbound calls. It indicates that enough of the dial string has been communicated for the switching function to connect the call to the Network Interface Device that will be associated with the called device in the external network.
2. The initial state of Initiated for D1C1 is used when D1 is “offhook” prior to the Make Call service request. The callID used for the Make Call shall be the same as the callID used for the initiated call.

17.1.18.3.2 Device-Type Monitoring Event Sequences

There are two types of Device-Type Monitoring Event sequences depending on the dialling sequence used for the called device.

- Case A: The dialling sequence for the called device is complete.

Table 17-99 Make Call—Device-Type Monitoring Event Sequences (Case A)

Monitored Device	Connection	Event	Event Cause
D1 (calling device)	D1C1	Service Initiated (optional)	Make Call, if calling device is prompted by the switching function.
			New Call, if the Make Call is done manually.
	D1C1	Originated	Normal
	D2C1	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (i.e., Forward, Do Not Disturb, etc.).(see item #1)	
D2 (called device)	D2C1	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (i.e., Forward, Do Not Disturb, etc.). (see item #1)	

- Case B: The dialling sequence for the called device is incomplete.

Table 17-100 Make Call—Device-Type Monitoring Event Sequences (Case B)

Monitored Device	Connection	Event	Event Cause
D1 (calling device)	D1C1	Service Initiated (optional)	Make Call, if calling device is prompted by the switching function.
			New Call, if the Make Call is done manually.
	D1C1	Digits Dialled	Normal
	D2C1	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (i.e., Forward, Do Not Disturb, etc.). (see item #1)	
D2 (called device)	D2C1	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (i.e., Forward, Do Not Disturb, etc.). (see item #1)	

17.1.18.3.3 Call-Type Monitoring Event Sequences

- Case A: The dialling sequence for the called device is complete.

Table 17-101 Make Call—Call-Type Monitoring Event Sequences (Case A)

Monitored Call	Connection	Event	Event Cause
C1	D1C1	Service Initiated (optional)	Make Call, if calling device is prompted by the switching function.
			New Call, if the Make Call is done manually.
	D1C1	Originated	Normal
	D2C1	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (i.e., Forward, Do Not Disturb, etc.). (see item #1)	

- Case B: The dialling sequence for the called device is an incomplete sequence.

Table 17-102 Make Call—Call-Type Monitoring Event Sequences (Case B)

Monitored Call	Connection	Event	Event Cause
C1	D1C1	Service Initiated (optional)	Make Call, if calling device is prompted by the switching function.
			New Call, if the Make Call is done manually.
	D1C1	Digits Dialled	Normal
	D2C1	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (i.e., Forward, Do Not Disturb, etc.).(see item #1)	

1. Providing of these events by the switching function does not affect the service completion conditions (i.e., a service is considered complete without these events being provided).

17.1.18.3.4 Functional Requirements

1. Prompting of the calling device during the processing of the Make Call service is switching function specific (display flashing, ring pattern, lamp blinking, etc.).
2. If the validation of the Make Call service fails for any reason, the computing function receives a negative acknowledgement, and no valid ConnectionIDs will have been created.
3. If the autoOriginate parameter is supported by the switching function and has a value of “Do Not Prompt”, and if the calling device is not capable of automatically answering the device, then the processing of this parameter is switching function dependent.
4. Call Forwarding and Do Not Disturb features for the calling device are not honoured for the call being made. If a switching function supports prompting of the calling device and detects that a feature was activated while processing the Make Call service and this feature cannot be overridden, then the switching function will return a negative acknowledgement.
5. All active features are honoured for the called device.
6. If a Make Call service is issued for a calling device that has a connection in the Initiated state (i.e., the computing function got a Service Initiated event for a phone manually going off-hook), and the switching function can support the issuing of the Make Call service under this condition, then the Connection ID in the positive acknowledgement of the Make Call service will contain the same Connection ID received on the Service Initiated event. If the switching function does not support the Make Call service under this condition, then the service will be rejected with a negative acknowledgement. (This note is an exception to requirement #2 in 9.5.1 on page 73.)
7. The calledDirectoryNumber parameter may contain a device identifier of null (a null formatted device identifier) or contain a Diallable Digits format (DD) device identifier that has a partial dialling sequence character (i.e., the “;” character) at the end of it. When this parameter is used in this manner, the computing function is indicating that it wishes to stage the dialling sequence. The completion of the dialling sequence can be accomplished either by entering the rest of the sequence manually at the actual device or the computing function can use the Dial Digits service to complete the sequence. The other types of calledDirectoryNumber parameter shall contain a complete dialling sequence. The switching function may have a timeout period for multi-stage dialling. If the dialling sequence does not complete prior to this timeout, it may either abort the call or attempt to use the digits already dialled and signal that dialling is complete with an originated event.
8. If the Make Call service is used to initiate a multistage dialling sequence, the computing function is signalled to continue the dialling sequence via either the Service Initiated event (i.e., if the calledDirectoryNumber is Null) or the Digits Dialled event (i.e., if the calledDirectoryNumber has a partial dialling sequence character).

9. When prompting a device which has a call appearance, bridged, or hybrid device configuration, only one of the appearances will be delivered the call.
10. The switching function may or may not adjust the digital data characteristics (e.g., connection rate) and connection information (e.g., number of channels) that were supplied on the Make Call service request (Use the capabilities exchange services to determine which feature the switching function supports). If the switching function supports adjusting the characteristics/connection information but cannot adjust them in this case, the service request will be rejected with an appropriate error code, if the original characteristics/ connection information cannot be provided. If the switching function can adjust of the characteristics/ connection information, the positive acknowledgement will contain the adjusted value or values. If the computing function determines that the adjusted values are not adequate, it can terminate the digital data call (e.g., Clear Call).
11. If the computing function makes a digital data call and wants to also bind a particular Media Service to the call, then the computing function shall use the Media Attach Service service.
12. The callingDevice parameter in the Make Call service request may represent a group of devices from which the actual calling device is selected. In this case, the callingDevice parameter in subsequent events shall refer to the actual calling device. In addition, the Originated event contains a parameter (originatingDevice) that represents the group of devices (the callingDevice parameter passed in service request).

17.1.19 Make Predictive Call

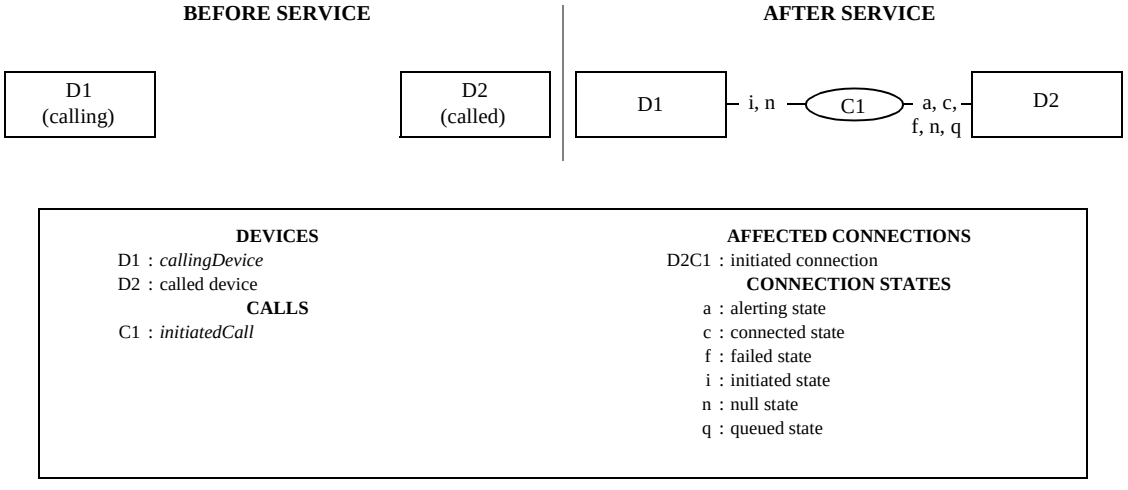
C → S

The Make Predictive Call service shall originate a call between two devices by first creating a connection to the called device. The service returns a positive acknowledgement that provides the connection at the called device.

Subsequent actions are taken depending upon the call progress and the actions requested. Examples are:

- An attempt is made to connect at the calling device because a connected state was determined at the called device.
- The call was cleared because a connected state was determined at the called device and the switching function detected a FAX and was instructed by the computing function to clear the call in this case.

Figure 17-22 Make Predictive Call Service



17.1.19.1 Service Request

Table 17-103 Make Predictive Call—Service Request

Parameter Name	Type	M/ O/C	Description
callingDevice	DeviceID	M	Specifies the device on behalf of which the call is originated.
calledDirectoryNumber	DeviceID	M	Specifies the called device.

signallingDetection	Structure	O	Specifies the switching function actions to be taken when the specified call progress conditions at the called device or network are met. It includes the following components:
			• signallingCondition (M) Enumerated - Specifies the conditions that must be satisfied before the switching function processes the signallingConditionsAction. The complete set of possible values is:
			  • callDelivered - Process the specified actions when the call is delivered to or answered at the called device (whichever occurs first).
			  • callEstablished - Process the specified actions when the call has been answered at the called device.
			• signallingConditionsAction (M) Enumerated - Action to be taken by the switching function upon detecting a signallingCondition. The complete set of possible values is:
			  • destinationDetection - The conditions/actions specified in the destinationDetection parameter should be followed.
			  • remainConnected - The called device is to remain in the call and the connection to the calling device is attempted.

Table 17-103 Make Predictive Call—Service Request (continued)

Parameter Name	Type	M/ O/C	Description

destinationDetection	List of Structures	O	Specifies the switching function actions to be taken when the specified signallingCondition at the called device is met. Each entry in the list contains the following:
			• destinationCondition (M) Enumerated - Specifies the condition that must be satisfied before the switching function processes the detectionAction. The complete set of possible values is:
			  • humanVoice - Process the specified actions when a human voice is detected at the called device.
			  • answeringMachine - Process the specified actions when an answering machine is detected at the called device.
			  • facsimileMachine - Process the specified actions when a (FAX) machine or modem is detected at the called device.
			• detectionAction (M) Enumerated - Action to be taken by the switching function upon detecting of a particular destination condition. The complete set of possible values is:
			  • clearCalledConnection - The connection for the called device is to be cleared.
			  • remainConnected - The called device is to remain in the call and the connection to the calling device is attempted.
			Note that if multiple entries are provided, the switching function shall process the detectionAction associated with the first destinationCondition that is detected.

defaultAction	Enumerated	O	Specifies the actions to be taken when the switching function determines that the condition specified in the signallingCondition component cannot be met. The complete list of possible values is: • clearCalledConnection - The connection for the called device is to be cleared. • remainConnected - The called device is to remain in the call and the connection to the calling device is attempted.
accountCode	AccountInfo	O	Specifies the account code to associate with the new call.
authCode	AuthCode	O	Specifies the authorization code to allow the call.
autoOriginate	Enumerated	O	Specifies if the calling device's connection is automatically answered (hands-free mode). The complete set of possible values is: • prompt (default) • doNotPrompt (auto originate)
alertTime	Value	O	Specifies the amount of time, in seconds, the called device is to be alerted before the call is cleared.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (Priority call, for example) to be associated with the call. See 12.2.4, "CallCharacteristics", on page 84 for the complete set of possible values. If the supported characteristics cannot be honoured, the switching function shall reject the service request.
userData	UserData	O	Specifies the user data to be sent to parties in the call.

Table 17-103 Make Predictive Call—Service Request (continued)

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information.

17.1.19.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.19.2.1 Positive Acknowledgement

Table 17-104 Make Predictive Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
initiatedCall	ConnectionID	M	Specifies the initial connection between the called device and the call. The ConnectionID shall have the CallID of the resulting new call and the DeviceID representing the called device.
initiatedCallInfo	ConnectionInformation	O	Specifies the connection information associated with the initiatedCall connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information.

17.1.19.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.19.3 Operational Model

17.1.19.3.1 Connection State Transitions

Table 17-105 Make Predictive Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1	Null	Initiated, Null
D2C1 (initiated)	Null	Alerting, Connected, Failed, Queued, or Null

17.1.19.3.2 Device-Type Monitoring Event Sequences

Table 17-106 Make Predictive Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D2 (called device) or the device representing D2 when D2 is outside the switching sub-domain.	D2C1	The sequence of events for the called device will depend on the type of called device and if it is in the switching sub-domain. See item #1 for the service completion criteria.	
	D1C1	Delivered (see item #2).	Normal, Entering Distribution

Table 17-106 Make Predictive Call—Device-Type Monitoring Event Sequences (continued)

Monitored Device	Connection	Event	Event Cause
D1 (calling device)	D1C1	Service Initiated (Optional) - see Functional Requirement #10 and Item #3).	Make Predictive Call, Reserved
	D2C1	The sequence of events for the called device will depend on the type of called device and if it is in the switching sub-domain. See item #1 for the service completion criteria.	
	D1C1	Delivered (see item #2).	Normal, Entering Distribution

17.1.19.3.3 Call-Type Monitoring Event Sequences

Table 17-107 Make Predictive Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D1C1	Service Initiated (Optional - see functional requirements #10 and Item #3).	Make Predictive Call, Reserved
	D2C1	The sequence of events for the called device will depend on the type of called device and if it is in the switching sub-domain See item #1 for the service completion criteria.	
	D1C1	Delivered (See item #2).	Normal, Entering Distribution

1. The completion criteria for this service depends upon the signallingCondition component specified in the service request:

   a. the Delivered or Established event (whichever flows first) for D2 if the signallingCondition is Delivered

   b. the Established event for D2 if the signallingCondition is Established

   In the case where the switching function detects that the conditions specified in the signallingCondition component cannot be met (for example when the call fails or queues at the called device), the service is completed when the switching function generates the event corresponding to the condition at the called device that prevented the signallingCondition from being met (Failed, Queued events, for example) or Connection Cleared if the switching function cleared the called connection (see Functional Requirement #4).

2. If the signallingConditionAction parameter specifies remainConnected, then depending on the type of calling device and its state, the Delivered, Queued, Failed, and Established events can flow for D1. These events are not part of the service completion criteria.

3. In the case where the switching function reserves the calling device for allocating the predictive call to it, Functional Requirement #10 applies but the cause code of reserved shall be used.

17.1.19.3.4 Functional Requirements

1. The service does not guarantee the connection to the calling device; only that an attempt will be made to connect to the device.

2. The calling device in a Make Predictive Call service request is often a group or ACD device that then allocates the new call to another calling device.

3. A typical use of the Make Predictive Call service is to place calls outside the CSTA switching sub-domain. The CSTA connection identifier reported in the positive acknowledgement refers to the Network Interface Device associated with the called device.

4. The parameter defaultAction is used in situations where the switching function detects that the signallingCondition in the service request cannot be met. For example, the requestor of the service has the capability to instruct the switching function to either connect the calling device for listening to the voice channel or to clear the call.

5. If the validation of the Make Predictive Call fails initial for any reason, the computing function receives a negative acknowledgement, and no valid ConnectionIDs will have been created.
6. If the autoOriginate parameter is supported by the switching function and has a value of attempt, and if the calling device is not capable of automatically answering the device, then the processing of this parameter is switching function dependent.
7. Call Forwarding and Do Not Disturb features for the calling device are not honoured for the call being made. If a switching function determines that it will not be able to deliver the call to the calling device (due to an active feature that cannot be overridden), then the switching function shall return a negative acknowledgement before the call is launched to the called device.
8. All active features are honoured for the called device.
9. Multi-stage dialling is not supported for the Make Predictive Call service.
10. The switching function may indicate that the calling device is involved with a predictive call prior to any call activity at the calling device by generating a Service Initiated event (with a cause of Make Predictive Call) for the calling device. When the Service Initiated event is generated in this case, it is generated prior to any call control events at the called device. If the switching function does not generate the Service Initiated event in this case, no call control events will be generated for the calling device until after the calling device is joined into the call. If the calling device cannot be joined in the call, a Connection Cleared event is generated for the initiatedConnection.

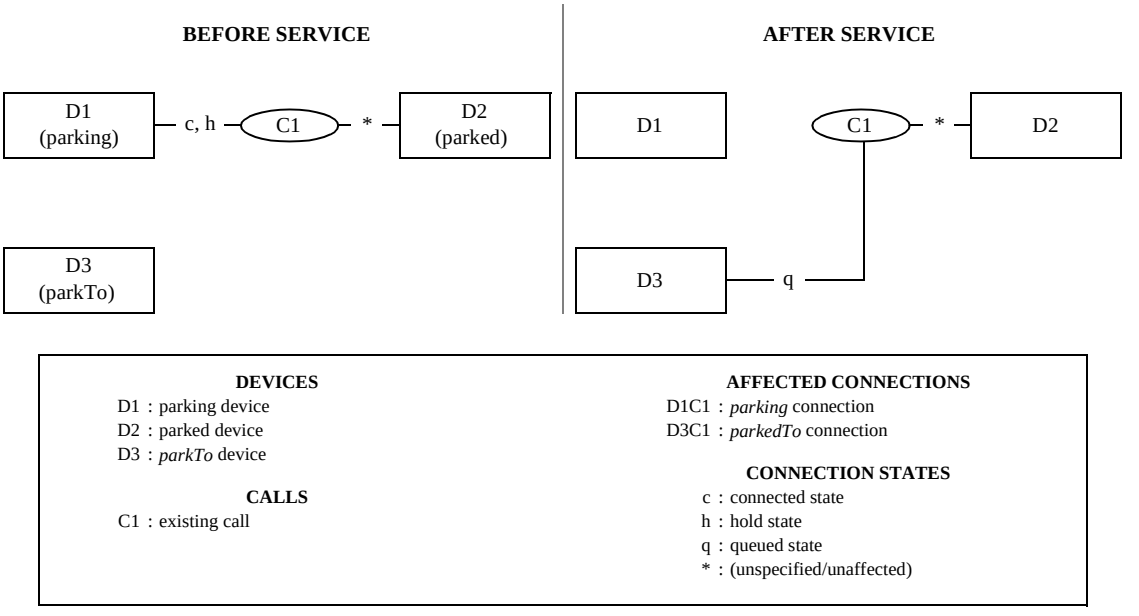
17.1.20 Park Call

C ⟶ S

The Park Call service moves a specified call at a device to a specified (parked-to) destination.

The device on whose behalf Park Call was invoked (the parking device) is no longer associated with the call (except when the parking device parks a call back to the parking device).

Figure 17-23 Park Call Service



Note that the parking device and the parkTo device may be the same (when a call is parked to the parking device).

Note that D1C1 can transit to the Queued state for certain Shared Bridged device configurations (see Functional Requirement #6).

17.1.20.1 Service Request

Table 17-108 Park Call—Service Request

Parameter Name	Type	M/ O/C	Description
parking	ConnectionID	M	Specifies the connection to be parked.
parkTo	DeviceID	M	Specifies the device to which the call is to be parked (parked-to device). This may be either a station or a Park device.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.20.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.20.2.1 Positive Acknowledgement

Table 17-109 Park Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
parkedTo	ConnectionID	O	Specifies the connection at the parkedTo device.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.20.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.20.3 Operational Model

17.1.20.3.1 Connection State Transitions

Table 17-110 Park Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (parking)	Hold or Connected	Null, Queued (see item #1)
D2C1 (parked)	(Unspecified)	(Unaffected; no transition due to this service).
D3C1 (parkTo)	Null	Queued

1. Connection D1C1 may transit to the Queued state for certain shared bridged device configurations (see functional requirement #5).

17.1.20.3.2 Device-Type Monitoring Event Sequences

Table 17-111 Park Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (parking device)	D1C1	Diverted (see items #1, #2)	Park or Normal
D2 (parked device)	D1C1	Diverted (optional) (see items #1, #2, #4)	Park or Normal
	D3C1	Queued	Park or Normal
D3 (parkTo device)	D3C1	Queued	Park or Normal

17.1.20.3.3 Call-Type Monitoring Event Sequences

Table 17-112 Park Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D1C1	Diverted (optional) (see items #1, #2, #5)	Park or Normal
	D3C1	Queued	Park or Normal

1. A Connection Cleared event will not be provided for the parking ConnectionID. The Diverted event implies the ConnectionID has gone to Null.
2. The Diverted event is not provided when a call is parked to the parking device (devices D1 and D3 are the same).
3. If the parking device stays offhook and receives busy or blocked tone, the switching function sends a Failed event (event cause of Blocked or Busy) for a call with a different callID (not C1), followed by a Connection Cleared event.

4. The switching function provides the Diverted event for D2 only if it is providing Diverted events for all devices in a call.This is indicated through the capabilities exchange services.
5. For call-type monitoring, the switching function provides the Diverted event for C1 only if it is providing Diverted events for call-type monitors. This is indicated through the capabilities exchange services.
6. If a Bridged event is generated for an Independent Shared Bridged device configuration (see Functional Requirement #5), it is not part of the service completion criteria.

17.1.20.3.4 Functional Requirements

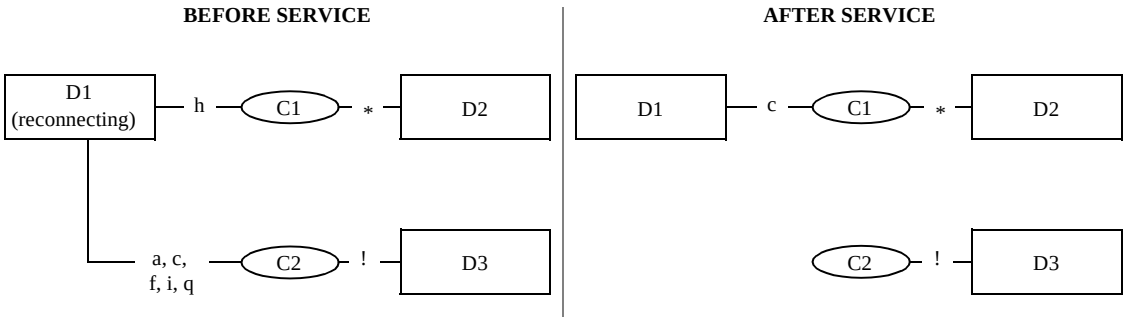
1. As a result of the Park Call service, the call ID associated with this call remains unchanged.
2. A call may be parked at a device which is already involved with the call. In this case the parking device (D1) and the parkTo (D3) are the same. Refer to the Operational Model description for specific requirements for this configuration.
3. The switching function may have a time-out period for the call once it is parked. If the call is not unparked before the time-out period ends, then the parked call may ringback the parking device or another switching defined device such as an attendant console, for example, or the call may be dropped.
4. Features such as Do Not Disturb and Forwarding are honoured at the parkedTo device when the call is being parked.
5. For Shared Bridged device configurations, when a call is parked from an appearance (parking) all appearances will be cleared from the call (i.e., Connection Cleared events with a cause of Normal Clearing), except when the call that is being parked is part of an Independent Shared Bridged device configuration and the appearance that is parking the call is not the last appearance connected into the call. In this case the appearance parking the call will return to the inactive mode (i.e., Bridged event with a cause of Normal) and the other appearances are unaffected. For more information, refer to Annex A.2.3, “Shared-Bridged”.
6. This service request does *not* support the use of a null device identifier or a Diallable Digits format (DD) device identifier that has a partial dialling sequence character (i.e., the “;” character) in it for the parkTo parameter. A complete and valid device identifier shall be provided, otherwise the switching function will reject the service request with a negative acknowledgement.

17.1.21 Reconnect Call

C → S

The Reconnect Call service will clear a specified connection at the reconnecting device and retrieve a specified held connection at the same device.

Figure 17-24 Reconnect Call Service



DEVICES
D1 : reconnecting device
D2 : device in call C1 with device D1
D3 : device in call C2 with device D3

CALLS
C1 : held call
C2 : active call

AFFECTED CONNECTIONS
D1C1 : *heldCall* connection

CONNECTION STATES
a : alerting state
c : connected state
f : fail state
h : hold state
i : initiated state
q : queued state
! : (unspecified)
* : (unspecified/unaffected)

17.1.21.1 Service Request

Table 17-113 Reconnect Call—Service Request

Parameter Name	Type	M/ O/C	Description
activeCall	ConnectionID	M	Specifies the connection to be cleared.
heldCall	ConnectionID	M	Specifies the held connection.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.21.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.21.2.1 Positive Acknowledgement

Table 17-114 Reconnect Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.21.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.21.3 Operational Model

17.1.21.3.1 Connection State Transitions

Table 17-115 Reconnect Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (heldCall)	Hold	Connected
D1C2 (activeCall)	Alerting, Connected, Initiated, Fail, or Queued	Null
D2C1	(Unspecified)	(Unaffected)
D3C2	(Unspecified)	(Unspecified)

17.1.21.3.2 Device-Type Monitoring Event Sequences

Table 17-116 Reconnect Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (reconnecting device)	D1C2	Connection Cleared	Normal Clearing, Call Canceled
	D1C1	Retrieved	Normal
D2 (and other devices in call C1)	D1C1	Retrieved	Normal
D3 (and other device in call C2)	D1C2	Connection Cleared	Normal Clearing, Call Canceled

17.1.21.3.3 Call-Type Monitoring Event Sequences

Table 17-117 Reconnect Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C2	D1C2	Connection Cleared	Normal Clearing, Call Canceled
C1	D1C1	Retrieved	Normal Clearing

17.1.21.3.4 Functional Requirements

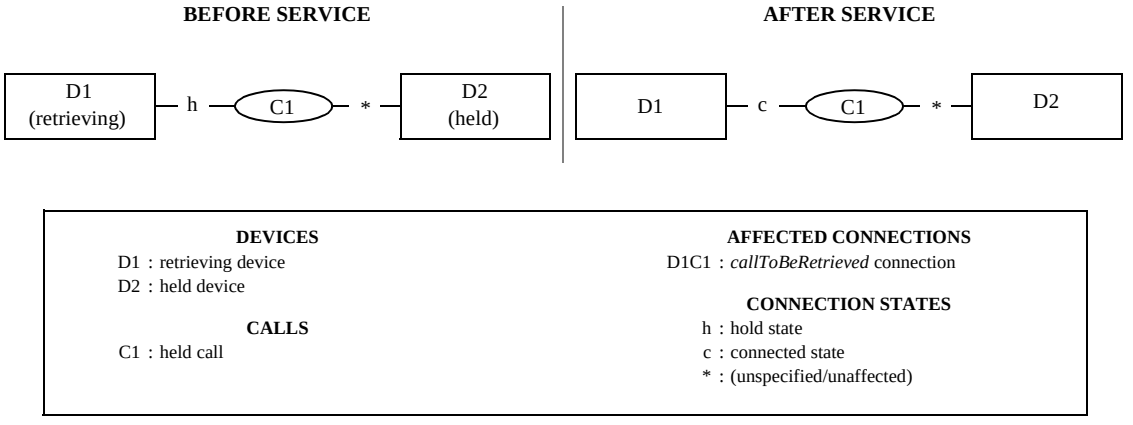
1. This service is a compound service that is equivalent to a computing function issuing a Clear Connection service for the activeCall ConnectionID and then issuing a Retrieve Call service for the heldCall ConnectionID.
2. If all appearances of a shared bridged device configuration are in the hold state and the heldCall parameter contains an appearance's connection ID in the call, then the other appearances in the device configuration will return to the inactive mode (queued state, Bridged events).
3. When the last connected appearance, in a shared bridged device configuration, is cleared as a result of the Reconnect service, all other device configuration appearance associated with the activeCall connection are also cleared from the call (i.e., Connection Cleared events).
4. If the Hold Call, Consultation Call, or the Alternate Call service was used to place the call on hold and the connectionReservation parameter was used to reserve the media stream channel(s) for the held call (e.g., ISDN bearer channel), then the same media stream channel(s) will be used for the call when it becomes reconnected.

17.1.22 Retrieve Call

C → S

The Retrieve Call service connects a specified held connection.

Figure 17-25 Retrieve Call Service



17.1.22.1 Service Request

Table 17-118 Retrieve Call—Service Request

Parameter Name	Type	M/ O/C	Description
callToBeRetrieved	ConnectionID	M	Specifies the held connection to be retrieved.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.22.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.22.2.1 Positive Acknowledgement

Table 17-119 Retrieve Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.22.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.22.3 Operational Model

17.1.22.3.1 Connection State Transitions

Table 17-120 Retrieve Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (callToBeRetrieved)	Hold	Connected
D2C1 (and any other connections in call C1)	(Unspecified)	(Unaffected; no transition due to this service).

17.1.22.3.2 Device-Type Monitoring Event Sequences

Table 17-121 Retrieve Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (retrieving device)	D1C1	Retrieved	Normal
D2 (and any other devices in call C1)	D1C1	Retrieved	Normal

17.1.22.3.3 Call-Type Monitoring Event Sequences

Table 17-122 Retrieve Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D1C1	Retrieved	Normal

17.1.22.3.4 Functional Requirements

1. The Retrieve Call service may only be requested for a connection at a device's physical element if there is no other connected connection already present at the device's physical element. A device's physical elements can only interact with one voice call at a time so if an attempt is made to retrieve a call while another voice call is connected, the switching function sends a negative acknowledgement to the Retrieve Call service request.
2. If the Hold Call, Consultation Call, or the Alternate Call service was used to place the call on hold and the connectionReservation parameter was used to reserve the media stream channel(s) for the held call (e.g., ISDN bearer channel), then the same media stream channel(s) will be used for the call when it becomes reconnected.
3. There may or may not be a media stream channel(s) available for the call on hold. In the case where a media stream channel(s) is not available, the switching function is unable to comply with the service request. In this case the switching function will send either a negative acknowledgement or the Service Completion Failure event, depending upon the acknowledgement model.
4. If all appearances of a shared-bridged or exclusive-bridged device configuration are in the hold state and the callToBeRetrieved parameter contains an appearance's connection ID in the call, then the other appearances in the device configuration will return to the inactive mode (queued state, Bridged events) in the case of shared-bridging or the blocked mode (fail state, failed events) in the case of exclusive bridging.

17.1.23 Single Step Conference Call

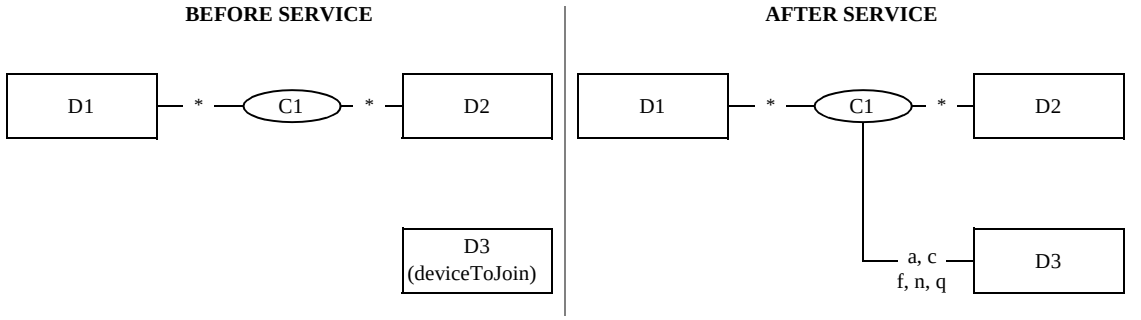
C → S

The Single Step Conference Call joins a new device into an existing call.

This service can be repeated to make n-device conference calls (subject to switching function limits).

This service is distinguished from the Join Call service by the way the device being added to the call perceives the direction of the resulting connection (e.g. alerts). In the case of the Join Call service, the device generates an outgoing connection (e.g. prompts). This affects the parameters associated with the service and the events flowing as a result of the service.

Figure 17-26 Single Step Conference Call Service



DEVICES
D1 : device in call C1 with device D2
D2 : device in call C1 with device D1
D3 : *deviceToJoin* device

CALLS
C1 : existing call

AFFECTED CONNECTIONS
D3C1 : *conferencedCall* connection

CONNECTION STATES
a : alerting state
c : connected state
f : fail state
n : null state
q : queued state
* : (unspecified/unaffected)

Note that the activeCall connection in the service request may be either D1C1 or D2C1.

17.1.23.1 Service Request

Table 17-123 Single Step Conference Call—Service Request

Parameter Name	Type	M/ O/C	Description
activeCall	ConnectionID	M	Specifies an existing connection in the call to which a new device is to be added.
deviceToJoin	DeviceID	M	Specifies the device that is to be added to the call.
participationType	Enumerated	O	Specifies the type of participation the added device has in the resulting call. The complete set of possible values is: • Active (default) - The added device actively participates in the resulting conference call. As a result, the flow direction of the deviceToJoin's connection (i.e., conferencedCall) will be Transmit and Receive. • Silent - The added device can listen but cannot actively participate in the resulting conference call. As a result, the flow direction of the deviceToJoin's connection (i.e., conferencedCall) will be Receive.
accountCode	AccountInfo	O	Specifies the account code to associate with the call.
authCode	AuthCode	O	Specifies the authorization code to allow the call.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
userData	UserData	O	Specifies the user data to be sent to parties in the call.

Table 17-123 Single Step Conference Call—Service Request (continued)

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.23.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.23.2.1 Positive Acknowledgement

Table 17-124 Single Step Conference Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
conferencedCall	ConnectionID	M	Specifies the callID of the existing active call and the DeviceID of the joining device.
conferencedCallInfo	ConnectionInformation	O	Specifies the connection information associated with the conferencedCall connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.23.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.23.3 Operational Model

17.1.23.3.1 Connection State Transitions

Table 17-125 Single Step Conference Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1	(Unspecified)	(Unaffected; no transition due to this service)
D2C1	(Unspecified)	(Unaffected; no transition due to this service)
D3C1 (conferencedCall)	Null	Alerting, Connected, Queued, Fail, or Null (Null if call moves away from D3 [i.e., forwarded]).

17.1.23.3.2 Device-Type Monitoring Event Sequences

Table 17-126 Single Step Conference Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1	D3C1 (conferencedCall)	*Note:* It is switching function dependent if other events flow before the Conferenced event. (See item #2)	
	D3C1 (conferencedCall)	Conferenced	Silent or Active Participation, or Single Step Conference
	D3C1 (conferencedCall)	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (See item #3)	
D2	D3C1 (conferencedCall)	*Note:* It is switching function dependent if other events flow before the Conferenced event. See Functional Requirement #2.	
	D3C1 (conferencedCall)	Conferenced	Silent or Active Participation, or Single Step Conference
	D3C1 (conferencedCall)	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (See Item #3)	
D3 (deviceToJoin)	D3C1 (conferencedCall)	*Note:* It is switching function dependent if other events flow before the Conferenced event. See Functional Requirement #2.	
	D3C1 (conferencedCall)	Conferenced	Silent or Active Participation, or Single Step Conference
	D3C1 (conferencedCall)	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (See Item #3)	

17.1.23.3.3 Call-Type Monitoring Event Sequences

Table 17-127 Single Step Conference Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D3C1	*Note:* It is switching function dependent if other events flow before the conferenced event. See Functional Requirement #2.	
	D3C1	Conferenced	Silent or Active Participation, or Single Step Conference
	D3C1 (conferencedCall)	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (see Item #3)	

1. The event cause is based on the value of the participationType parameter.
   - If it was set to active, then the event cause will be Active Participation.
   - If it was set to silent, then the event cause will be Silent Participation.
   - If the switching function cannot determine the type used, then the event cause will be Single Step Conference.
2. If the deviceToJoin is not in the connected state after the Single Step Conference Call service is executed, then normal call progress events will flow regarding the state of this device after the Conferenced event. Devices in the resulting call will hear the call progress associated with the deviceToJoin.
3. Providing of these events by the switching function does not affect the service completion conditions (i.e., a service is considered complete without these events being provided).

17.1.23.3.4 Functional Requirements

1. The call ID in the ConnectionIDs for the resulting conference call is the same as that in the original active call.
2. It is switching function dependent to determine at which point in the call progress a device is actually conferenced into a call. For example, a device may become part of an existing call before it is actually conferenced into a call. In this case, additional events may flow before the Conferenced event. The callID in the ConnectionIDs for these events shall be the same as that in the original active call and shall be the same as in the Conferenced event that follows.
3. The appearances in a shared bridged device configuration are unaffected by this service.
4. This service request does not support the use of a null device identifier or a Diallable Digits format (DD) device identifier that has a partial dialling sequence character (i.e., the ";" character) in it for the deviceToJoin parameter. A complete and valid device identifier shall be provided, otherwise the switching function will reject the service request with a negative acknowledgement.
5. For the new destination device (deviceToJoin), all active features for the device are honoured.
6. If this service is used to add a device into a digital data call, the connection for the deviceToJoin (i.e., conferencedCall) will inherit the characteristics of the call with the exception of connection flow direction and the number of channels used. In these cases, the connection flow direction is dependent on the value of participationType and the number of channels is based on what the switching function will use to allow the deviceToJoin to effectively participate in the call.

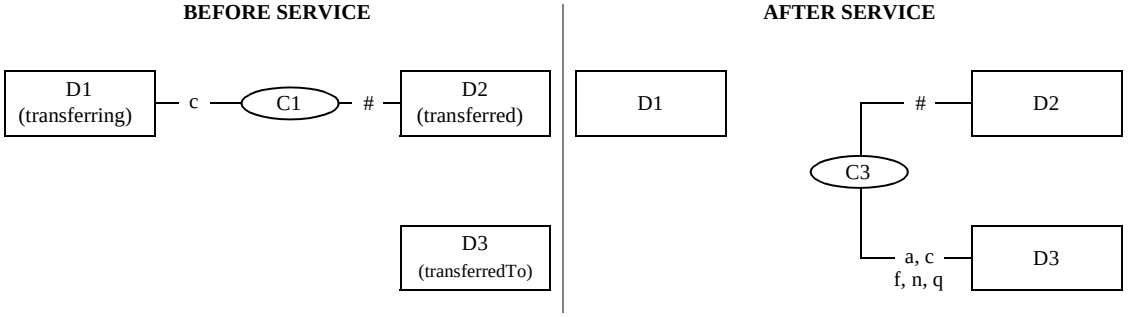
17.1.24 Single Step Transfer Call

C → S

The Single Step Transfer Call service transfers an existing connected connection at a device to another device.

This transfer is performed in a single-step, that is the device doing the transfer does not have to place the existing call on hold before issuing the Single Step Transfer Call service.

Figure 17-27 Single Step Transfer Call Service



DEVICES
D1 : transferring device
D2 : transferred device
D3 : *transferredTo* device

CALLS
C1 : *activeCall*
C3 : *transferredCall*

AFFECTED CONNECTIONS
D1C1 : *activeCall* connection
D3C3 : *transferredCall* connection

CONNECTION STATES
a : alerting state
c : connected state
f : fail state
n : null state
q : queued state
: (unspecified/inherited)

17.1.24.1 Service Request

Table 17-128 Single Step Transfer Call—Service Request

Parameter Name	Type	M/ O/C	Description
activeCall	ConnectionID	M	Specifies the connected connection in the call to be replaced.
transferredTo	DeviceID	M	Specifies the new called (transferred-to) device.
accountCode	AccountInfo	O	Specifies the account code to associate with the call.
authCode	AuthCode	O	Specifies the authorization code to allow the call.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
userData	UserData	O	Specifies the user data to be sent to parties in the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.24.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.24.2.1 Positive Acknowledgement

Table 17-129 Single Step Transfer Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
transferredCall	ConnectionID	M	Specifies the ConnectionID of the deviceToTransferTo in the transferred call.
connections	ConnectionList	O	Specifies information on each device/connection that is remaining in the call as a result of the service.
transferredCallInfo	ConnectionInformation	O	Specifies the connection information associated with the transferredCall connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.24.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.24.3 Operational Model

17.1.24.3.1 Connection State Transitions

Table 17-130 Single Step Transfer Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (activeCall)	Connected	Null
D2C1 and any other Connections in the call C1	(Unspecified)	Null
D2C3	Null	(Inherited from the corresponding connection in call C1)
D3C3 (transferredCall)	Null	Alerting, Connected, Queued, Fail, or Null (Null if call moves away from the originally transferred-to device [i.e., forwarded])

17.1.24.3.2 Device-Type Monitoring Event Sequences

Table 17-131 Single Step Transfer Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (transferring)	D1C1	Transferred (see item #3)	Single Step Transfer
D2 (transferred device) (and any other devices in the resulting call)	D1C1	Transferred	Single Step Transfer
	D3C3 (transferredCall)	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (see item #4)	
D3 (transferredTo)	D3C3 (transferredCall)	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (see item #4)	

17.1.24.3.3 Call-Type Monitoring Event Sequences

Table 17-132 Single Step Transfer Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1	D1C1	Transferred (see item #3)	Single Step Transfer
C3	D3C3 (transferredCall)	Events depend on the type of device (ACD, extension, external device, etc.) and features activated at it at the time of this service (Forward, Do Not Disturb, etc.). (see item #4)	

1. If the transferredTo device is not in the connected state after the Single Step Transfer Call service is executed, then normal call progress events will flow regarding the state of this device after the Transferred event.
2. A Connection Cleared event will not be seen for the activeCall ConnectionID; the Transferred event implies that the connection at the transferringDevice has been cleared.
3. If the transferring device stays off-hook and receives busy or blocked tone, the switching function sends a Failed event (event cause of Blocked or Busy) for a call with a different callID (not C1 or C3), followed by a Connection Cleared event.
4. Providing of these events by the switching function does not affect the service completion conditions (i.e., a service is considered complete without these events being provided).
5. If a Bridged event is generated for an Independent Shared Bridged device configuration (see Functional Requirement #5), it is not part of the service completion criteria.

17.1.24.3.4 Functional Requirements

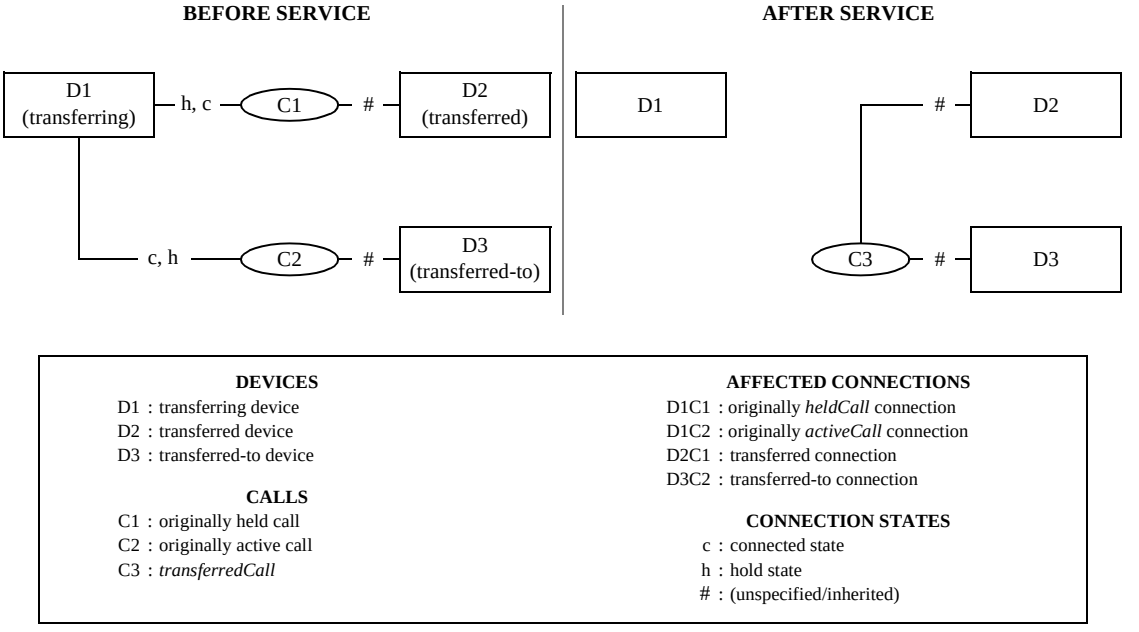
1. The computing function should never assume the reuse of callIDs, although some switching functions may reuse the activeCall callID for the transferredCall callID
2. As a result of this service, the device or devices being transferred may hear call-progress signals for the new device. For example, if the new device answers, the call resumes or if the new device is busy or does not answer, the other devices in the call may receive resulting call progress tones. In either case, the transferring device has left the call and has no further relationship with it.
3. If the Single Step Transfer Call service fails for any reason after the positive acknowledgement is returned, the switching function may return the call to the transferring device. Refer to 6.8.3, “Recall”, on page 55.
4. For the new destination device (deviceToTransferTo), all active features for the device are honoured.
5. Shared Bridged device configurations, when a call is transferred by an appearance (transferring device) all appearances will be cleared from the call (i.e., Connection Cleared events with a cause of Normal Clearing), except when the call that is being transferred is part of an Independent Shared Bridged device configuration and the appearance that is transferring the call is not the last appearance connected into the call. In this case the appearance transferring the call will return to the inactive mode (i.e., Bridged event with a cause of Normal) and the other appearances are unaffected. For more information, refer to Annex A.2.3, “Shared-Bridged”.
6. This service request does not support the use of a null device identifier or a Diallable Digits format (DD) device identifier that has a partial dialling sequence character (i.e., the “;” character) in it for the transferredTo parameter. A complete and valid device identifier shall be provided, otherwise the switching function will reject the service request with a negative acknowledgement.
7. If this service is used to single step transfer a digital data call, the connection for the transferredTo device (i.e., transferredCall) will inherit the characteristics of the call.

17.1.25 Transfer Call

C ⟶ S

The Transfer Call service transfers a call held at a device to an active call at the same device. The held and active calls at the transferring device shall be merged into a new call. Also, the Connections of the held and active calls at the transferring device shall become Null and their ConnectionIDs shall be released (i.e., the transferring device is no longer involved with the call).

Figure 17-28 Transfer Call Service



Note that both the D1C1 and D1C2 connections may be held or connected prior to the transfer.

17.1.25.1 Service Request

Table 17-133 Transfer Call—Service Request

Parameter Name	Type	M/ O/C	Description
heldCall	ConnectionID	M	Specifies the held connection.
activeCall	ConnectionID	M	Specifies the active connection.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.25.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

17.1.25.2.1 Positive Acknowledgement

Table 17-134 Transfer Call—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
transferredCall	ConnectionID	M	Specifies the ConnectionID of the transferred-to device in the resulting call.
connections	ConnectionList	O	Specifies information on each device/connection that is remaining in the call as a result of the service.

Table 17-134 Transfer Call—Positive Acknowledgement (continued)

Parameter Name	Type	M/ O/C	Description
transferredCallInfo	ConnectionInformation	O	Specifies the connection information associated with the transferredCall connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.1.25.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

17.1.25.3 Operational Model

17.1.25.3.1 Connection State Transitions

Table 17-135 Transfer Call—Connection State Transitions

Connection	Initial State (Required)	Final State
D1C1 (heldCall)	Connected, Hold	Null
D1C2 (activeCall)	Connected, Hold	Null
D2C1 and any other connections in call C1	(unspecified)	Null
D3C2 and any other connections in call C2	(unspecified)	Null
D3C3 and any other connections in call C3	Null	(Inherited from the corresponding connection in calls C1 and C2)

17.1.25.3.2 Device-Type Monitoring Event Sequences

Table 17-136 Transfer Call—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
D1 (transferring)	(See item #4)	Transferred (see item #3)	Transfer or Normal
D2 (transferred)	(See item #4)	Transferred	Transfer or Normal
D3 (transferred-to)	(See item #4)	Transferred	Transfer or Normal

17.1.25.3.3 Call-Type Monitoring Event Sequences

Table 17-137 Transfer Call—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
C1 or C2	(See item #4)	Transferred (see items #1 and #2)	Transfer

1. If the transferred-to device was not in the connected state prior to the Transfer Call service, then normal call progress events will flow after the Transferred event. Devices in the original held call will hear the call progress associated with the connection to the transferred-to device.
2. A Connection Cleared Event will not be provided for the activeCall or heldCall ConnectionIDs. The Transferred event implies connection cleared for the transferring device.
3. If the transferring device stays off-hook and receives busy or blocked tone, the switching function may send a Failed event (event cause of Blocked or Busy) for a different call, followed by a Connection Cleared event.
4. There are multiple connections affected by this service.

5. If a Bridged event is generated for an Independent Shared Bridged device configuration (see Functional Requirement #4), it is not part of the service completion criteria.

17.1.25.3.4 Functional Requirements

1. To get the connections for each of the devices to their initial states, the computing function can either:
   - Use the Consultation Call service to place a call on hold and place a new call.
   - Use the Alternate Call service to place a call on hold and answer an alerting call.

   In either of these cases, if the switching function supports the consultOptions parameter in these services, the computing function shall provide this parameter with a value of either “Conference Only” or “Unrestricted”.

   Some switching function support a third approach for preparing for the Conference Call service involving the use of a hold service followed by a Make Call.

   The fourth approach supported by some switching functions involves two held calls at the same device.

   Certain switching functions also support a fifth approach involving two active calls at the same device.

   The computing function should use the capabilities exchange services to determine which of these approaches is supported by the switching function.
2. If the computing function uses the Consultation Call service and specifies the consultOptions parameter with a value of Conference, and then attempts to complete the consultation call with a Transfer Call service, it will be rejected with a negative acknowledgement.
3. If the call is not established at the transfer-to device, the switching function may return the call to the transferring device. Refer to 6.8.3, “Recall”, on page 55.
4. For Shared Bridged device configurations, when a call is transferred by an appearance (transferring device) all appearances will be cleared from the call (i.e., Connection Cleared events with a cause of Normal Clearing), except when the call that is being transferred is part of an Independent Shared Bridged device configuration and the appearance that is transferring the call is not the last appearance connected into the call. In this case the appearance transferring the call will return to the inactive mode (i.e., Bridged event with a cause of Normal) and the other appearances are unaffected. For more information, refer to Annex A.2.3, “Shared-Bridged”.
5. The computing function should never assume the reuse of callIDs, although some switching functions may reuse one or the other.

17.2 Events

Table 17-138 Call Control Events Summary

Call Control Event	Description	Pg.
17.2.1 Bridged	Indicates that an appearance at a shared bridged device configuration has been placed into an inactive mode (i.e., queued state).	268
17.2.2 Call Cleared	Indicates that all devices have been removed from an existing call.	270
17.2.3 Conferenced	Indicates that the conferencing device has conferenced itself or another device with an existing call.	273
17.2.4 Connection Cleared	Indicates that a device in a call has disconnected or dropped out from a call.	278
17.2.5 Delivered	Indicates that a call is being presented to a device in either the Ringing or Entering Distribution modes of the alerting state.	282
17.2.6 Digits Dialled	Indicates that a call or feature is being attempted from a device and that a portion of the dialling sequence has been completed.	286
17.2.7 Diverted	Indicates that a call has been diverted from a device.	289
17.2.8 Established	Indicates that a device has answered or has been connected to a call.	293
17.2.9 Failed	Indicates that a call cannot be completed and/or a connection has entered the Fail state.	297
17.2.10 Held	Indicates that an existing call has been put on hold.	302
17.2.11 Network Capabilities Changed	Indicates that a situation occurred during a call's progress in a public or private network that modifies its signalling capability (i.e., inter-networking).	304
17.2.12 Network Reached	Indicates that a call has been connected to an external network using a Network Interface Device (e.g., trunk, CO Line).	307
17.2.13 Offered	Indicates that a call is in a pre-delivery state at a device (prior to ringing indication or delivering ringback, for example).	311
17.2.14 Originated	Indicates that a call is being attempted from a device.	315
17.2.15 Queued	Indicates that a call has been queued.	318
17.2.16 Retrieved	Indicates that a previously held call has been retrieved.	322
17.2.17 Service Initiated	Indicates that a device has gone off-hook for service or is being prompted to go off-hook.	324
17.2.18 Transferred	Indicates that an existing call has been transferred to another device and that the device transferring the call has been dropped from the call.	327

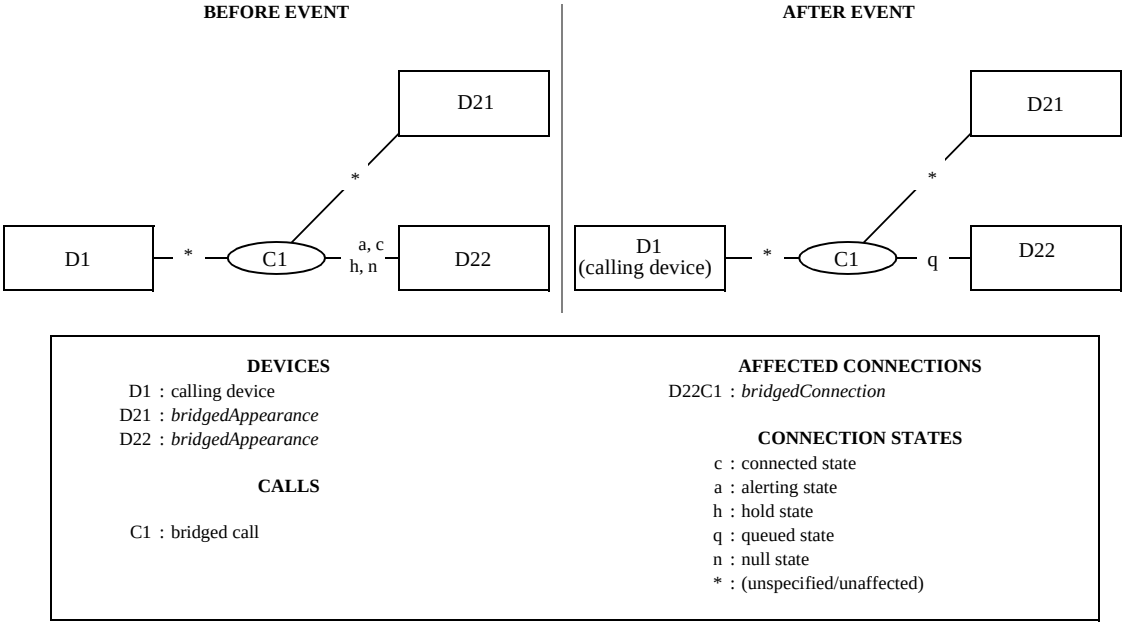
17.2.1 Bridged

The Bridged event indicates that an appearance at a shared bridged device configuration has been placed into an inactive mode (i.e., queued state). See the device configuration section for more details on shared bridged device configurations,

Common situations that generate this event include:

- When the first appearance in a shared bridged device configuration appearance connects into the call at the device (i.e., Answer Call) (manual and service request initiated), the other appearances enter the inactive mode.
- When the first appearance in a shared bridged device configuration leaves a call at the device and at least one appearance is still connected into the call. (i.e., Clear Connection) (manual and service request initiated).
- When the first appearance in a shared bridged device configuration retrieves a call and the other appearances return to the inactive mode.

Figure 17-29 Bridged Event



17.2.1.1 Event Parameters

Table 17-139 Bridged—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
bridgedConnection	ConnectionID	M	Specifies the connection of the appearance that was placed in the inactive mode.
bridgedAppearance	SubjectDeviceID	M	Specifies the appearance which was placed in the inactive mode.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the bridged appearance at the bridged device: Queued • For the other appearances at the bridged device & all other devices left in the call: (unaffected) This parameter is mandatory for events generated for device-type monitors and shall otherwise not be provided.

Table 17-139 Bridged—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
correlatorData	CorrelatorData	O	Specifies the correlator data associated with the call.
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 101 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
bridgedConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the bridgedConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies the non-standardized information attached to the event.

17.2.1.2 Event Causes

Table 17-140 Bridged—Event Causes

Event Cause	Description	Associated Features
Normal	An appearance has been placed into an inactive mode.	Multi-Appearance

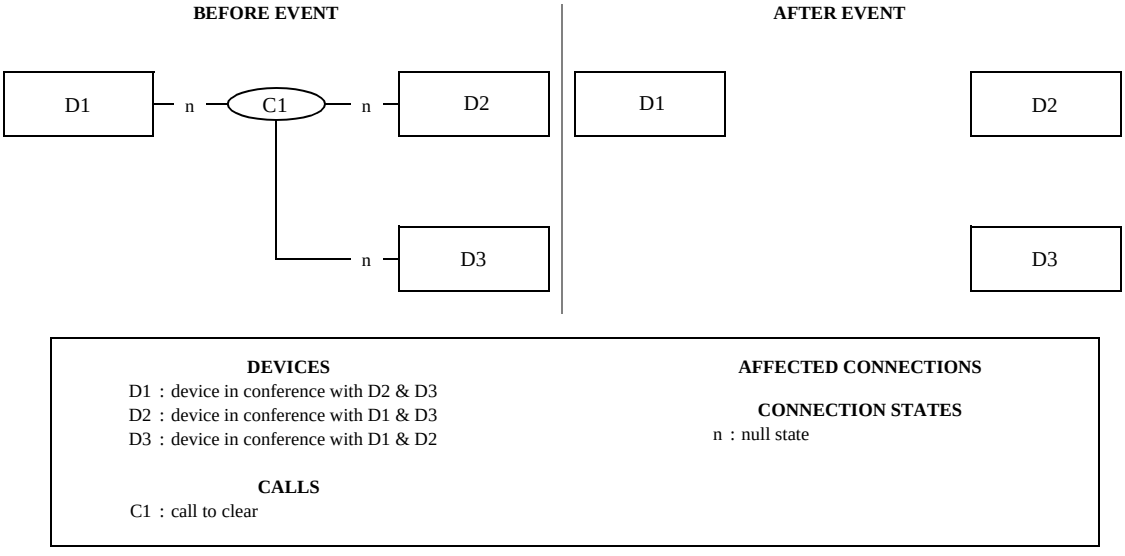
17.2.2 Call Cleared

The Call Cleared event is only provided for calls that are being call-type monitored. This event indicates that a call has been cleared and no longer exists within the switching sub-domain. A call is cleared when there is no longer any device associated with the call.

Common situations that generate this event include:

- After the last remaining device disconnects from the call.
- All devices in a call are immediately disconnected from a call such as when a conference controller dissolves a call.
- The computing function issues a Clear Call service request that is successful.

Figure 17-30 Call Cleared Event



17.2.2.1 Event Parameters

Table 17-141 Call Cleared—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
clearedCall	ConnectionID	M	Specifies the ConnectionID of the cleared call. Note that the DeviceID shall be omitted in this ConnectionID.
correlatorData	CorrelatorData	O	Specifies the correlator data associated with the call.
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
cause	EventCause	M	Specifies the reason for the event. The cause supplied will be the same as the cause supplied on the last Connection Cleared event for the call.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.

Table 17-141 Call Cleared—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information attached to the event.

17.2.2.2 Event Causes

Table 17-142 Call Cleared—Event Causes

Event Cause	Description	Associated Features
Busy	The call was cleared because it reached a busy destination.	Connection Failure
Call Back	The call was cleared after a call back or call back message feature was invoked.	Call Back Call-Related, Call Back Message Call-Related
Call Cancelled	The call was cleared without a device going on-hook (via the Clear Call service, for example).	Clear Connection, Clear Call, Connection Failure
Call Not Answered	The call was cleared because it was not answered before a timer elapsed.	Clear Connection, Clear Call, Connection Failure
Destination Out of Order	The call was cleared because it encountered a destination out of service.	Connection Failure
Do Not Disturb	The call was cleared because the call encountered a destination that has the Do Not Disturb feature set.	Do Not Disturb
Incompatible Destination	The call was cleared because it encountered an incompatible destination.	Connection Failure
Invalid Account Code	The call was cleared because of an invalid account code.	Connection Failure
Invalid Number Format	The call was cleared because of an incorrect dialled number.	Connection Failure
Maintenance	The call was cleared because it encountered a facility or endpoint in a maintenance condition.	Connection Failure
Network Congestion	The call was cleared because it reached a congested network	External Call
Network Not Obtainable	The call was cleared because it could not reach the destination network.	External Call
Network Out of Order	The call was cleared because it encountered a network that is out of order.	Connection Failure
Network Signal	The call was cleared that involved a device that is outside of the switching sub-domain.	Clear Connection, Clear Call, Connection Failure, External Call
Normal Clearing	The call was cleared (a more specific event cause cannot be provided).	Clear Connection, Clear Call, Any feature
Not Available Bearer Service	The call was cleared because it was requested with bearer capability that is currently not available.	Connection Failure
Not Supported Bearer Service	The call was cleared because it was requested with a bearer capability that is currently not supported.	Connection Failure
Number Changed	The call was cleared because the called number has been changed to a new number and the call cannot be completed.	Connection Failure
Number Unallocated	The call was cleared because the called number is not allocated to a subscriber.	Connection Failure
Override	The call was cleared because of an Override (e.g. Intrude) feature.	Connection Failure

Table 17-142 Call Cleared—Event Causes (continued)

Event Cause	Description	Associated Features
Reorder Tone	The call was cleared because the call encountered a reorder condition.	Connection Failure
Resource not Available	The call was cleared because resources were not available.	Connection Failure
Selected Trunk Busy	The call was cleared because the specific selected Network Interface Device (e.g., trunk, CO Line) is busy.	Connection Failure
Trunks Busy	The call was cleared because there was no available Network Interface Device (e.g. trunk, CO Line).	Connection Failure
Unauthorized Bearer Service	The call was cleared because it was requested with an unauthorized bearer capability.	Connection Failure

17.2.2.3 Functional Requirements

1. Connection Cleared events shall be sent for all devices in the call before the Call Cleared event is sent. The Call Cleared event can not be used as a substitute for the appropriate Connection Cleared events.
2. The Call Cleared event is only sent to call-type monitors. It is never sent to device-type monitors.

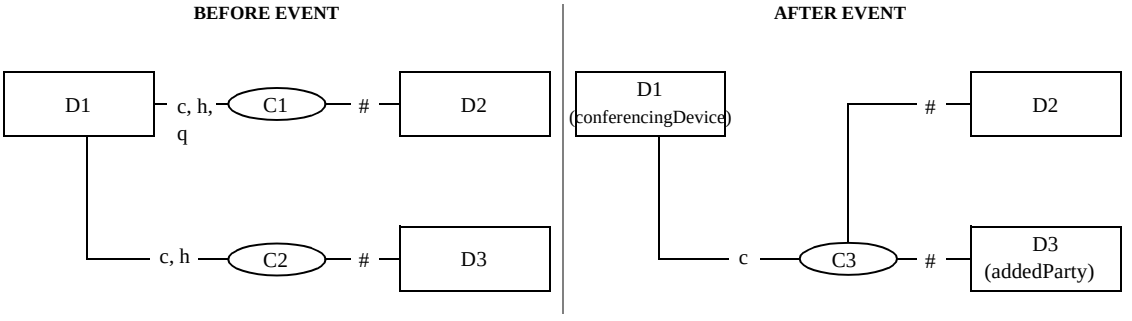
17.2.3 Conferenced

The Conferenced event indicates that the conferencing device has conferenced itself or another device with an existing call and that no devices have been removed from the resulting call.

Common situations that generate this event include:

- Two step conferencing situations (manual and service initiated)
- Single step conferencing situations (manual and service initiated)
- Intrude situations (manual and service initiated)
- Join situations (manual and service initiated)

Figure 17-31 Conferenced Event—Case A: Two Step Conferencing



DEVICES

D1 : *conferencingDevice* device
D2 : device in conference with D1 and D3
D3 : *addedParty* device

CALLS*

C1 : primary old call
C2 : secondary old call
C3 : resulting call

* the primary/secondary old call and the primary/secondary old call connections mentioned in this figure are from the perspective of the conferencingDevice (D1). See Functional Requirement #1.

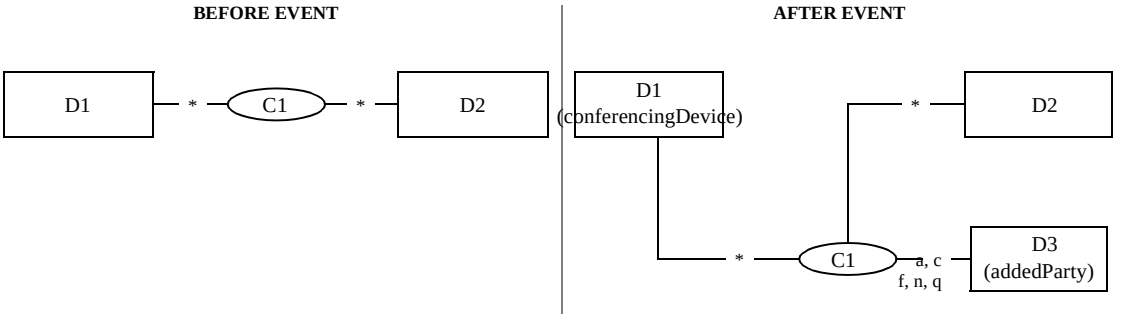
AFFECTED CONNECTIONS

D1C1 : *primaryOldCall* connection
D1C2 : *secondaryOldCall* connection
D2C1 : initial call at device D2
D3C2 : initial call at device D3
D1C3 : resulting call at device D1
D2C3 : resulting call at device D2
D3C3 : resulting call at device D3

CONNECTION STATES

c : connected state
h : hold state
q : queued state
: (unspecified/inherited)

Figure 17-32 Conferenced Event—Case B: Single Step Conferencing



DEVICES
D1 : *conferencingDevice* device
D2 : device in conference with D1 & D3
D3 : *addedParty* device

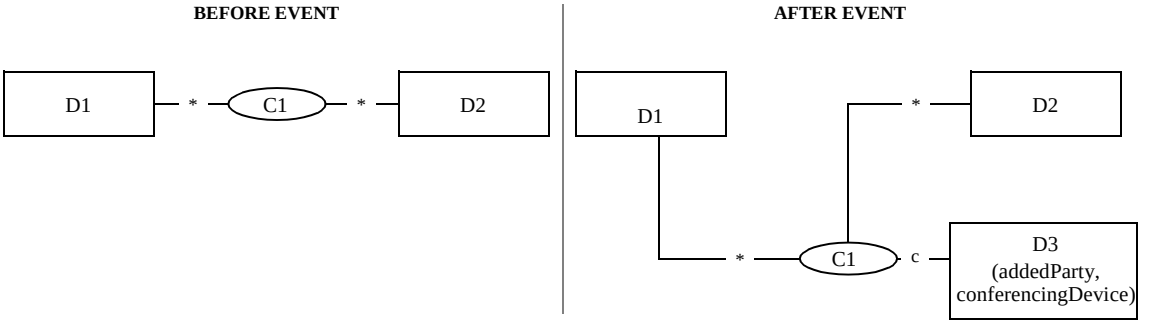
CALLS
C1 : primary old call
C1 : resulting call (1)

AFFECTED CONNECTIONS
D1C1 : resulting call at device D1
D2C1 : resulting call at device D2
D3C1 : resulting call at device D3

CONNECTION STATES
a : alerting state
c : connected state
f : failed state
n : null state
q : queued state
* : (unspecified/unaffected)

1. Since the Call ID does not change for a Single Step Conference Call, the primary old call and the resulting call are the same.

Figure 17-33 Conferenced Event—Case C: Join Call



DEVICES
D1 : device in conference with D2 & D3
D2 : device in conference with D1 & D3
D3 : *addedParty* device, *conferencingDevice*

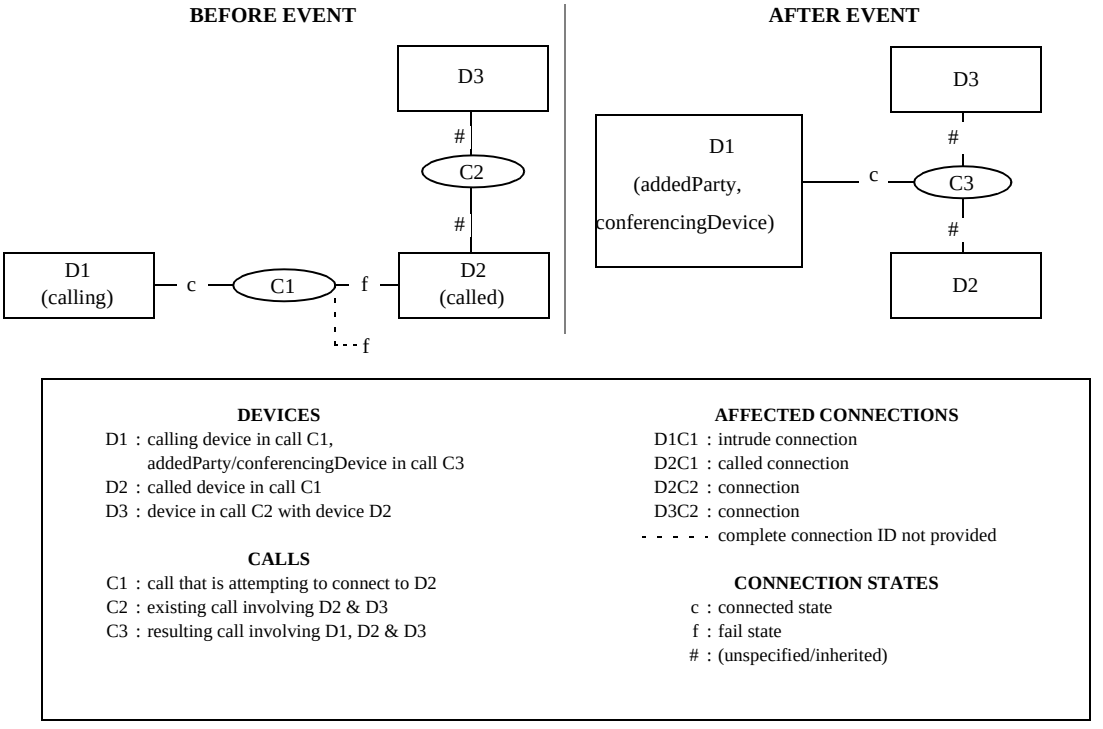
CALLS
C1 : primary old call
C1 : resulting call [1]

AFFECTED CONNECTIONS
D1C1 : resulting call at device D1
D2C1 : resulting call at device D2
D3C1 : resulting call at device D3

CONNECTION STATES
* : (unspecified/unaffected)
c : connected state

1. Since the Call ID does not change for the Join Call service, the primary old call and the resulting call are the same.

Figure 17-34 Conferenced Event - Case D: Intrude Call



Refer to 6.8.2, “Connection Failure”, on page 53 for a complete description of partial connection IDs.

Refer to 17.1.16, “Intrude Call”, on page 231 for information on how a connection can transit to the connected state.

17.2.3.1 Event Parameters

Table 17-143 Conferenced—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
primaryOldCall	ConnectionID	M	Specifies the connection of the primary call. See Functional Requirement #1.
secondaryOldCall	ConnectionID	C	Specifies the connection of the secondary call. If the switching function supports the “fixed-view” option (as indicated by the capability exchange services), this parameter is mandatory. If the switching function supports the “local-view” option, this parameter is mandatory if there are two known calls involved with the conference (before the conference is created) from the perspective of the monitored device, otherwise it shall not be provided. See Functional Requirement #1.
conferencingDevice	SubjectDeviceID [1]	M	Specifies the device ID of the conferencing device.
addedParty	SubjectDeviceID [1]	M	Specifies the device ID of the last device added to the call.
conferenceConnections	ConnectionList	M	Specifies information on each device/ConnectionID in the resulting conference call. See Functional Requirement #2.

Table 17-143 Conferenced—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices that are in the conference call: • Conferencing device - Connected (or Unaffected for Case B). • Other devices - (See above figures for specific cases of the Conferenced event) This parameter is mandatory for events generated for device-type monitors and otherwise shall not be provided.
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call and shall otherwise not be provided.
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information attached to the event.

1. Note that SubjectDeviceID refers to a parameter type—not the subject device of the Conferenced event. This parameter type is used to represent the two devices in this event because the two devices are affected by the generation of this event (i.e., the conferencing device and the addedParty device). However, there is only one device which is the subject of the event and that is the conferencing device. For more details on the SubjectDeviceID parameter type, see 12.3.29, “Subject-DeviceID”, on page 117.

17.2.3.2 Event Causes

Table 17-144 Conferenced—Event Causes

Event Cause	Description	Associated Features
Active Participation	The call was conferenced and the added device can actively participate in the call. This feature is typically used to allow intrusion by a supervisor with the ability to speak and listen into an agent call.	Single Step Conference Call, Intrude Call, Join Call
Conference	The call was conferenced because of a two step conference.	Conference Call, Conference
Join Call	The call was conferenced because of a Join Call.	Join Call

Table 17-144 Conferenced—Event Causes (continued)

Event Cause	Description	Associated Features
Network Signal	The call was involved in a conference located outside of the switching sub-domain.	External Call
Normal	The call was conferenced (a more specific event cause cannot be provided).	Conference
Override	The call was conferenced because of an override (e.g., Intrude Call) feature. This event cause may be used when the participation type (active, silent) cannot be provided.	Intrude Call
Silent Participation	The call was conferenced and the added device can silently participate in the call. This feature allows a third party, such as an ACD agent supervisor, to join the call. The joining party can hear the entire conversation, but cannot be heard by either originating party. The feature, sometimes called Silent Intrusion, may provide a tone to one or both parties to indicate that the joining party is listening to the call.	Single Step Conference Call, Intrude Call, Join Call
Single Step Conference	The call was conferenced because of a single step conference.	Single Step Conference Call, Intrude Call

17.2.3.3 Functional Requirements

1. The contents of the primaryOldCall and the secondaryOldCall parameters may be either a “fixed view” or a “local view” of the connections at a device before the conference has been completed. The switching function indicates which view it provides via the connectionView parameter in the capability exchange services.
    - fixed view - for each conferenced event generated by monitors placed on different devices in a call, the switching function provides the same information in the primaryOldCall and the secondaryOldCall parameters independent of the monitorType (call or device-type monitor) and independent of the role of the device in the conference (conferencingDevice, addedParty, etc.). The meaning of these parameters for the fixed-view are:
        - primaryOldCall - specifies the first call visible at the conferencingDevice.
        - secondaryOldCall - specifies the second call visible at the conferencingDevice.
    - local view - for each conferenced event generated by monitors placed on different devices in a call, the switching function provides different information in the primaryOldCall and the secondaryOldCall parameters that depends upon which call was made visible first, from the perspective of the monitored device. The meaning of these parameters for the local-view are:
        - primaryOldCall - specifies the first call visible at the monitored device. For example, for a device-type monitor placed on the conferencingDevice (two step conference), this is the first call placed on hold (C1). For a device-type monitor placed on the addedParty device, this is the first (and only) call involved in the conference (C2) from the perspective of the monitored device.
        - secondaryOldCall - specifies the second call visible at the monitored device. For example, for a device-type monitor placed on the conferencingDevice (two step conference), this is the consultation call (C2). For a monitor placed on the addedParty device, there is no secondaryOldCall parameter.
2. The conferenceConnections parameter is a list that contains the new ConnectionID and may contain the old ConnectionID, the DeviceID (values such as ANI, etc.), and for externally located devices the associated Network Interface DeviceID.
3. The computing function should never assume the reuse of callIDs, although some switching functions may reuse one or the other.
4. There may be other devices present in the resulting conference call which are not part of the conference.

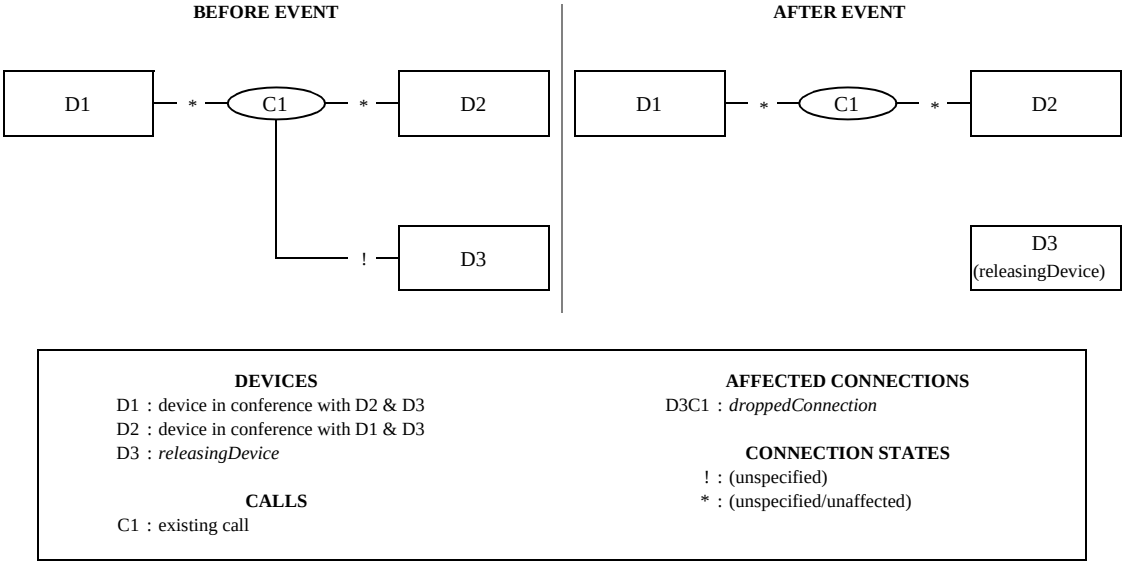
17.2.4 Connection Cleared

The Connection Cleared event indicates that a single device has disconnected or dropped out of a call.

Common situations that generate this event include:

- A user manually terminates the call (by going on-hook, for example).
- The Clear Connection service is successfully invoked.
- Connection clears as a result of some other service's operation.

Figure 17-35 Connection Cleared Event



17.2.4.1 Event Parameters

Table 17-145 Connection Cleared—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
droppedConnection	ConnectionID	M	Specifies the connection of the device that was dropped from the call.
releasingDevice	SubjectDeviceID	M	Specifies the device that dropped from the call.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the clearing device: Null • For the other devices left in the call: (unaffected) This parameter is mandatory for events generated for device-type monitors and otherwise shall not be provided.
correlatorData	CorrelatorData	O	Specifies the correlator data associated with the call.
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
chargingInfo	ChargingInfo	O	Specifies a total value of charging or currency units for the device that dropped from the call.
cause	EventCause	M	Specifies the reason for the event.

Table 17-145 Connection Cleared—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
droppedConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the droppedConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information attached to the event.

17.2.4.2 Event Causes

Table 17-146 Connection Cleared—Event Causes

Event Cause	Description	Associated Features
Alert Time Expired	The connection was cleared because a timer associated with the Make Predictive call expired.	Make Predictive Call
Busy	The connection was cleared because the call reached a busy destination.	Connection Failure
Call Back	The connection was cleared after a Call Back or a Call Back Message feature has been invoked.	Call Back Call-Related, Call Back Message Call-Related
Call Cancelled	The connection was cleared without a device going on-hook (via the Clear Call service, for example).	Clear Connection, Clear Call, Connection Failure
Call Not Answered	The connection was cleared because the call was not answered before a timer elapsed.	Clear Connection, Clear Call, Connection Failure
Destination Detected	The connection was cleared because the call encountered a specific destination.	Make Predictive Call
Destination Not Obtainable	The connection was cleared because the call could not reach the destination.	Connection Failure
Destination Out of Order	The connection was cleared because the call encountered a destination that is out of service.	Connection Failure
Do Not Disturb	The connection was cleared because the call encountered a destination that has the Do Not Disturb feature set.	Do Not Disturb
Incompatible Destination	The connection was cleared because the call encountered an incompatible destination.	Connection Failure
Invalid Account Code	The connection was cleared because of an invalid account code.	Connection Failure
Invalid Number Format	The connection was cleared because of an incorrect dialled number.	Connection Failure

Table 17-146 Connection Cleared—Event Causes (continued)

Event Cause	Description	Associated Features
Maintenance	The connection was cleared because it encountered a facility or endpoint in a maintenance condition.	Connection Failure
Network Congestion	The connection was cleared because the call reached a congested network.	External Call
Network Not Obtainable	The connection was cleared because the call could not reach the destination network.	External Call
Network Out of Order	The connection was cleared because the call encountered a network that is out of order.	Connection Failure
Network Signal	The device located outside the switching sub-domain has dropped from the call.	Clear Connection, Clear Call, Connection Failure, External Call
Normal Clearing	The connection was cleared (a more specific event cause cannot be provided).	Clear Connection, Clear Call, Any feature
Not Available Bearer Service	The connection was cleared because the call was requested with a bearer capability that is currently not available.	Connection Failure
Not Supported Bearer Service	The connection was cleared because the call was requested with a bearer capability that is currently not supported.	Connection Failure
Number Changed	The connection was cleared because the called number has been changed to a new number and the call cannot be completed.	Connection Failure
Number Unallocated	The connection was cleared because the called number is not allocated to a subscriber.	Connection Failure
Override	The connection was cleared because of an override (e.g., Intrude Call) feature.	Connection Failure
Queue Cleared	A call is queued in multiple ACD queues and the connection associated with one of the ACD queues was cleared because the call was distributed to an available agent from another ACD queue.	ACD
Reorder Tone	The connection was cleared because the call encountered a reorder condition. When this occurs, the network usually provides Reorder Tone to indicate that a request (call, feature, or supplementary service) was not recognizable. This condition typically results when a user dials a number that is not valid or attempts to obtain a service that is not enabled for that user or device.	Connection Failure
Resource Not Available	The connection was cleared because of resources not available.	Connection Failure
Selected Trunk Busy	The connection was cleared because the specific selected Network Interface Device (e.g., trunk, CO Line) is busy.	Connection Failure
Trunks Busy	The connection was cleared because there was no available Network Interface Device (e.g., trunk, CO Line).	Connection Failure
Unauthorized Bearer Service	The connection was cleared because the call was requested with an unauthorized bearer capability.	Connection Failure

17.2.4.3 Functional Requirements

1. A Connection Cleared event will be generated for appearances of a shared bridged device configuration under the following conditions.

- The call is ended.
- The appearance permanently drops from the call through some feature/service.
- The call moves away from the device configuration and none of the appearances are connected into the call.
- The call moves away from the device configuration and the device configuration is an appearance interdependent shared bridged device configuration.

The event cause will be Normal Clearing, the droppedConnection will be the connection identifier associated with the appearance and the releasingDevice will depend on the type of referencing model that is supported by the switching function (refer to 6.1.7, “Referencing Devices, Elements, Appearances and Device Configurations”, on page 36).

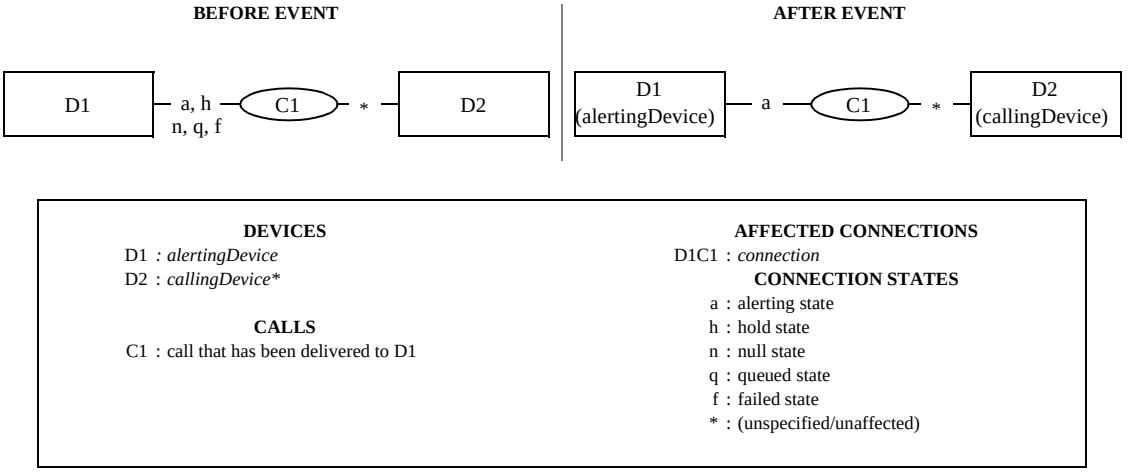
17.2.5 Delivered

The Delivered event indicates that a call is being presented to a device in either the Ringing or Entering Distribution modes of the alerting state.

Common situations that generate this event include:

- A call has been assigned to a device and that device is alerting.
- A call has been assigned to a distribution device such as an ACD, routeing device, or hunt group.

Figure 17-36 Delivered Event



*There are some situations where D1 can be the calling device (e.g. Make Predictive Call).

17.2.5.1 Event Parameters

Table 17-147 Delivered—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
connection	ConnectionID	M	Specifies the connection that is alerting.
alertingDevice	SubjectDeviceID	M	Specifies the device that is alerting.
callingDevice	CallingDeviceID	M	Specifies the calling device.
calledDevice	CalledDeviceID	M	Specifies the originally called device.
lastRedirectionDevice	RedirectionDeviceID	M	Specifies the previously known redirected from device.
originatingNIDConnection	ConnectionID	O	Specifies the connection of the Network Interface Device (NID) that the call originated from. If this parameter is supported, it shall be provided if there is an originating NID associated with the call, otherwise it shall not be provided. See Functional Requirement #4.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the alerting device: Alerting • For the calling device: (unaffected) This parameter is mandatory for events generated for device-type monitors and otherwise shall not be provided.

Table 17-147 Delivered—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call and shall otherwise not be provided.
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
networkCallingDevice	NetworkCallingDeviceID	O	Specifies the original calling device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
networkCalledDevice	NetworkCalledDeviceID	O	Specifies the original called device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
associatedCallingDevice	AssociatedCallingDeviceID	C	Specifies the Network Interface Device associated with the calling device if the call is an external incoming call. This parameter is mandatory for all external incoming calls and shall not be provided otherwise.
associatedCalledDevice	AssociatedCalledDeviceID	C	For outgoing external calls, this parameter specifies the Network Interface Device associated with the originally called device. For incoming external calls, this parameter specifies a device within the switching sub-domain associated with the originally called device. This parameter is mandatory for all external outgoing calls and it is optional for external incoming calls.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
connectionInfo	ConnectionInformation	O	Specifies the connection information associated with the alerting connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information attached to the event.

17.2.5.2 Event Causes

Table 17-148 Delivered—Event Causes

Event Cause	Description	Associated Features
ACD Busy	The call was delivered to an ACD device that has currently no available agents.	ACD, Consultation Call

Table 17-148 Delivered—Event Causes (continued)

Event Cause	Description	Associated Features
ACD Forward	The call was delivered to a new device and is no longer queued at the previous ACD device.	ACD
ACD Saturated	The call was delivered to an ACD device that has currently no internal resources available (including agents).	ACD, Consultation Call
Call Back	The call was delivered to a device because of a previously set call back feature.	Call Back Call-Related
Call Forward	The call was delivered to a device after it was forwarded (any type of forwarding).	Call Forwarding
Call Forward—Busy	The call was delivered to a device after it was forwarded because of a busy condition.	Call Forwarding
Call Forward—Immediate	The call was delivered to a device after it was forwarded (forwarding on all conditions).	Call Forwarding
Call Forward—No Answer	The call was delivered to a device after it was forwarded because of a no answer condition.	Call Forwarding
Distributed	The call was delivered to a device because the call has moved from the distribution mechanism (ACD, Hunt) to an available device associated with the distribution mechanism.	ACD, Routeing Services
Entering Distribution	The call was delivered to a distribution mechanism (ACD, Hunt) for the purpose of being distributed.	ACD, Routeing Services
Key Operation	The call was delivered to a device that has an appearance in a call appearance, bridged, or hybrid device configuration.	Multiple Appearance
Multiple Alerting	The call that is alerting the subject device is also alerting other devices.	Multiple Alerting
Network Signal	The call was delivered to a device that is outside of the switching sub-domain.	External Calls
New Call	The call was not redirected.	Any feature
No Available Agents	The call was delivered to a device because there were no available devices in a group (ACD).	ACD, Forwarding
Normal	The call was delivered to a device (a more specific cause cannot be provided).	Any feature
Overflow	The call was delivered to a device after it overflowed a queue, group, or target.	ACD
Override	The call was delivered to a device as a result of an override (e.g., Intrude Call) feature.	Intrude Call
Recall	The call was delivered to a device as part of the recall feature (e.g., due to a time-out associated with a feature that failed to complete).	Recall
Redirected	The call was delivered to a device after it was diverted from or deflected to this device.	Deflect Call, Divert Call
Remains in Queue	The call was delivered to a device while the calling device's position in the queue is retained. For example, the call could be delivered to a VRU or other automated answering equipment while the calling device retains its position in the ACD queue.	ACD, Queuing, Single Step Conference, Join Call
Resources not available	The call was delivered because some resources were not available (ACD, forwarding).	ACD, Forwarding

Table 17-148 Delivered—Event Causes (continued)

Event Cause	Description	Associated Features
Single Step Transfer	The call alerted the device due to a Single Step Transfer service.	Single Step Transfer
Single Step Conference	The call alerted the device due to a Single Step Conference service.	Single Step Conference
Timeout	The event was generated because a trunk timer expired. It is not generated as the result of a particular network event or condition.	External calls

17.2.5.3 Functional Requirements

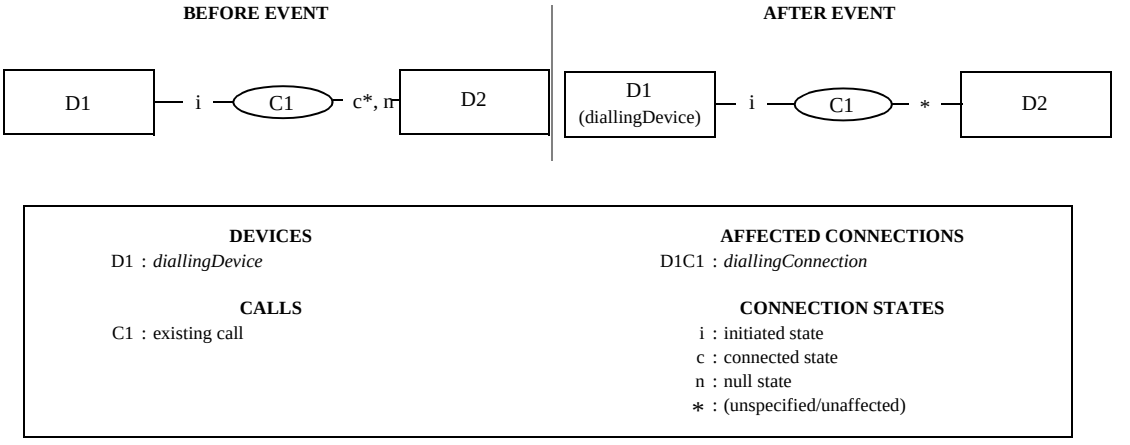
1. The Delivered event should not be used to determine when a device is physically ringing. (The Ringer Status event should be used for this.)
2. When a call is delivered to a bridged device configuration, multiple Delivered events are sent (one for each appearance of the device configuration). Each event will contain the following type of information:
   - The event for the appearance which is delivered the call first will have the appropriate event cause on why the call is being delivered to the device configuration. All subsequent events for the other appearances will have an event cause of Key Operation
   - The connection parameter will be the connection identifier of the specific appearance.
   - The alertingDevice will depend on the type of referencing model that is supported by the switching function (refer to 6.1.7, “Referencing Devices, Elements, Appearances and Device Configurations”, on page 36).
   - The calledDevice or associatedCalledDevice will contain the device identifier associated with the logical element of the device.
3. When observing a call or a device in a call, and the call diverts from a device in the call, if the switching function does not provide the Diverted event for all devices in a call or for call-type monitors (as indicated through the capabilities exchange services option), the computing function shall use the alertingDevice, calledDevice, lastRedirectionDevice, and cause parameters to properly track the progress of the call as the result of the redirection. See 6.8.6, “Tracking a Diverted Call”, on page 57 for more information.
4. If there is more than one calling device (i.e. a conference calling back to a device), then the originatingNIDConnection parameter will not be present.
5. For external incoming calls, the contents of the networkCallingDeviceID and the networkCalledDeviceID parameters shall not change as long as the associatedCallingDevice remains in the call. This differs from the callingDeviceID and the calledDeviceID parameters in that these parameters may change as a result of features such as transfer, forwarding, and conferencing, etc.

17.2.6 Digits Dialled

The Digits Dialled event indicates that a call or feature is being attempted from a device and that a portion of the dialling sequence has been completed. It implies that only part of the input activity is complete, and that more of the dialling sequence needs to be supplied before the entire input activity is complete.

After the entire dialling sequence is complete, the Originated event will be generated in the case where a call is being attempted. The last Digits Dialled event received prior to receiving the Originated event will report the last digits dialled to complete the dialling sequence.

Figure 17-37 Digits Dialled Event



Note that the *connected state when D2 is a NID.

17.2.6.1 Event Parameters

Table 17-149 Digits Dialled—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
diallingConnection	ConnectionID	M	Specifies the connection at which the digits were dialled.
diallingDevice	SubjectDeviceID	M	Specifies the device at which the digits were dialled.
diallingSequence	DeviceID	M	Specifies the sequence of digits that was dialled.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the device dialling the digits: initiated. This parameter is mandatory for events generated for device-type monitors and otherwise shall not be provided.
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call and shall otherwise not be provided.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
networkCallingDevice	NetworkCallingDeviceID	O	Specifies the original calling device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.

Table 17-149 Digits Dialled—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
networkCalledDevice	NetworkCalledDeviceID	O	Specifies the original called device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
associatedCallingDevice	AssociatedCallingDeviceID	C	Specifies the Network Interface Device associated with the calling device if the call is an external incoming call. This parameter is mandatory for all external incoming calls and shall not be provided otherwise.
associatedCalledDevice	AssociatedCalledDeviceID	C	For outgoing external calls, this parameter specifies the Network Interface Device associated with the originally called device. For incoming external calls, this parameter specifies a device within the switching sub-domain associated with the originally called device. This parameter is mandatory for all external outgoing calls and it is optional for external incoming calls.
diallingConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the diallingConnection connection. If this parameter is not present, then the connection information is switching function specific.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.2.6.2 Event Causes

Table 17-150 Digits Dialled—Event Causes

Event Cause	Description	Associated Features
Conference	Digits were dialled as part of a consultation for the purpose of a conference.	Consultation Call with consultOptions of ConferenceOnly
Consultation	Digits were dialled as part of a consultation.	Consultation Call
Network Dialling	Digits were dialled after a call has left the switching sub-domain.	External Calls
Normal	Digits were dialled (a more specific event cause cannot be provided).	Any feature, Dial Digits, Make Call
Transfer	Digits were dialled as part of a consultation for the purpose of a transfer.	Consultation Call with consultOptions of TransferOnly

17.2.6.3 Functional Requirements

1. This event will only be generated when the computing function has a device-type or call-type monitoring applied to a device that is initiating a call.
2. The grouping and number of digits reported in each event is switching function dependent.
3. The first Digits Dialled event includes the first digit specified in the Make Call or Consultation Call service which started the dialling sequence. If no digits were specified in the initiating Make Call or Consultation Call service (in the case where a null string was passed in the service request), the diallingSequence parameter in the Digits Dialled event contains a null string.

4. The digit sequence reported by the sequence of Digits Dialled events reflects the digits actually dialled through manual interaction or by CSTA service requests. It will not reflect any filtering or manipulation of the digit sequence by the switching function
5. If a Network Reached event is received prior to the receipt of an originated event, it may not be possible to determine the complete dialling sequence.
6. The switching function may have a timeout period for multi-stage dialling. If the dialling sequence does not complete prior to this timeout, it may either abort the call or attempt to use the digits already dialled and signal that dialling is complete with an originated event.

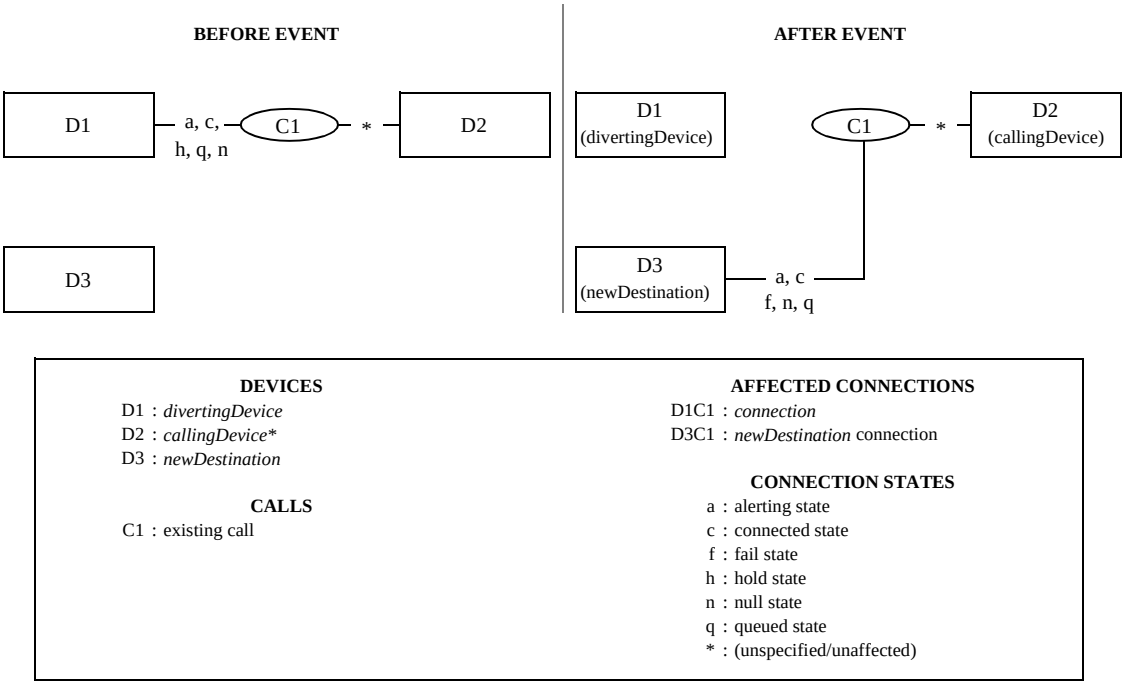
17.2.7 Diverted

The Diverted event indicates that a call has been diverted from a device and that the call is no longer present at the device.

Common situations that generate this event include:

- A call leaves a device that has some type of forwarding feature activated. Examples are Ring No Answer, Forward Immediate (depends upon the forwarding feature model supported by the switching function. See 6.8.1, “Forwarding”, on page 51), Recall, etc.
- A call leaves an ACD/Split/Hunt group device to be redirected to an agent, an extension, another ACD/Split/Hunt Group device, or to an offsite destination.
- A call leaves an ACD queue and is redirected to either an agent or an extension.
- A divert (Deflect, Pick, etc.) is successfully invoked.

Figure 17-38 Diverted Event



*There are some situations where D1 can also be the Calling Device (e.g. Make Predictive Call).

17.2.7.1 Event Parameters

Table 17-151 Diverted—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
connection	ConnectionID	M	Specifies the connection that was diverted.
divertingDevice	SubjectDeviceID	M	Specifies the device from which the call was diverted.
newDestination	SubjectDeviceID	M	Specifies the device to which the call was diverted.

Table 17-151 Diverted—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
callingDevice	CallingDeviceID	C	Specifies a device remaining in the call with the newDestination device. This parameter shall be provided in the case of Immediate Forwarding, where forwarding is triggered after the call is delivered to a device (see Function Requirement #3). Otherwise the parameter is optional.
calledDevice	CalledDeviceID	C	Specifies the originally called device. This parameter shall be provided in the case of Immediate Forwarding, where forwarding is triggered after the call is delivered to a device (see Function Requirement #3). Otherwise the parameter is optional.
lastRedirectionDevice	RedirectionDeviceID	M	Specifies the previously known redirected from device. Note that this is not the divertingDevice that caused this event, but the device that performed a redirection towards the current divertingDevice just before the current redirection.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the diverting device: Null • For the other devices left in the call: (unaffected) This parameter is mandatory for events generated for device-type monitors and otherwise shall not be provided.
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call and shall otherwise not be provided.
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
connectionInfo	ConnectionInformation	O	Specifies the connection information associated with the connection. If this parameter is not present, then the connection information is switching function specific.
networkCallingDevice	NetworkCallingDeviceID	O	Specifies the original calling device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
networkCalledDevice	NetworkCalledDeviceID	O	Specifies the original called device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.

Table 17-151 Diverted—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
associatedCallingDevice	AssociatedCallingDeviceID	C	Specifies the Network Interface Device associated with the calling device if the call is an external incoming call. This parameter is mandatory for all external incoming calls and shall not be provided otherwise.
associatedCalledDevice	AssociatedCalledDeviceID	C	For outgoing external calls, this parameter specifies the Network Interface Device associated with the originally called device. For incoming external calls, this parameter specifies a device within the switching sub-domain associated with the originally called device. This parameter is mandatory for all external outgoing calls and it is optional for external incoming calls.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information attached to the event.

17.2.7.2 Event Causes

Table 17-152 Diverted—Event Causes

Event Cause	Description	Associated Features
ACD Forward	The call was diverted from one ACD device to a new ACD device and is no longer queued at the previous ACD device.	ACD
Call Forward	The call was diverted from a device after it was forwarded from another device (any type of forwarding).	Call Forwarding
Call Forward—Busy	The call was diverted from a device after it was forwarded (forwarding on a busy condition).	Call Forwarding
Call Forward—Immediate	The call was diverted from a device after it was forwarded (forwarding on all conditions).	Call Forwarding
Call Forward—No Answer	The call was diverted from a device after it was forwarded because of a no answer condition.	Call Forwarding
Call Not Answered	The call was diverted from a device because it was not answered before a timer elapsed.	Recall
Call Pickup	The call was diverted from a device because of the pickup feature.	Directed Pickup Call, Group Pickup Call
Distributed	The call was diverted from a device because it was distributed by a distribution group (ACD or Hunt group).	ACD
Do Not Disturb	The call was diverted from a device because the device had the Do Not Disturb feature set.	Call Forwarding
No Available Agents	The call was diverted from a device because there were no available devices in a group (ACD).	ACD
Normal	The call was diverted from a device (a more specific cause cannot be provided).	Any feature
Overflow	The call was diverted from a device after it overflowed a queue, group, or target.	ACD
Park	The park feature has been invoked to either park or un-park a call at a device.	Park Call

Table 17-152 Diverted—Event Causes (continued)

Event Cause	Description	Associated Features
Recall	The call was diverted from a device as part of a recall feature (e.g., due to a time-out associated with a feature that failed to complete).	Recall
Redirected	The call was diverted from a device because of a deflect or divert feature.	Deflect Call, Divert Call
Resources Not Available	The call was diverted from a device because there were no resources available to process it.	ACD

17.2.7.3 Functional Requirements

1. Based upon the switching function (as indicated through the capabilities exchange services), the switching function may send the Diverted event to either:
   - Only the divertingDevice (D1) for device-type monitors, and not for call-type monitors, or
   - To all devices in a call (device-type monitors) and for call-type monitors.
2. When observing a call or a device in a call, and the call previously diverts from a device in the call, if the switching function does not provide the Diverted event for all devices in a call or for call-type monitors (as indicated through the capabilities exchange services option), the computing function shall use the divertingDevice, calledDevice, lastRedirectionDevice, and cause parameters to properly track the progress of the call as a result of the previous redirection. See 6.8.6, “Tracking a Diverted Call”, on page 57 for more information.
3. In the case of Immediate Forwarding, where forwarding is triggered *after* the call is delivered to a device (as indicated by the capability exchange services), the Diverted event for the Diverting device shall contain a cause of Immediate Forwarding and a localConnectionInfo of Null. Since this Diverted event is the first and last event generated for the call at the Diverting device, it is an example of a Null to Null connection state transition.
4. For external incoming calls, the contents of the networkCallingDeviceID and the networkCalledDeviceID parameters shall not change as long as the associatedCallingDevice remains in the call. This differs from the callingDeviceID and the calledDeviceID parameters in that these parameters may change as a result of features such as transfer, forwarding, and conferencing, etc.

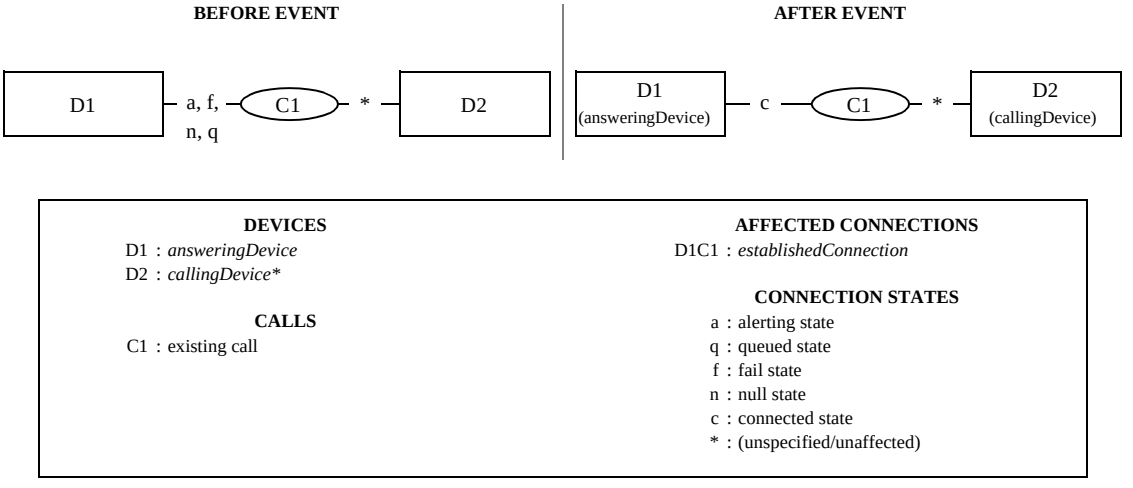
17.2.8 Established

The Established event indicates that a call has been answered at a device or that a call has been connected to a device.

Common situations that generate this event include:

- A call has been answered at a device (e.g., a user has manually gone off-hook).
- The Answer Call service has been successfully invoked.
- A call has been picked by another device.

Figure 17-39 Established Event



*There may be some situations where D1 can be the calling device (e.g. Make Predictive Call).

17.2.8.1 Event Parameters

Table 17-153 Established—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
establishedConnection	ConnectionID	M	Specifies the connection that was connected.
answeringDevice	SubjectDeviceID	M	Specifies the device that connected into the call.
callingDevice	CallingDeviceID	M	Specifies the calling device.
calledDevice	CalledDeviceID	M	Specifies the originally called device.
lastRedirectionDevice	RedirectionDeviceID	M	Specifies the previously known redirected from device.
originatingNIDConnection	ConnectionID	O	Specifies the connection of the Network Interface Device (NID) that the call originated from. If this parameter is supported, it shall be provided if there is an originating NID associated with the call, otherwise it shall not be provided. See Functional Requirement #4.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the answering device: Connected • For the calling device: (unaffected - never Null) This parameter is mandatory for events generated for device-type monitors and otherwise shall not be provided.

Table 17-153 Established—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call and shall otherwise not be provided.
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
networkCallingDevice	NetworkCallingDeviceID	O	Specifies the original calling device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
networkCalledDevice	NetworkCalledDeviceID	O	Specifies the original called device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
associatedCallingDevice	AssociatedCallingDeviceID	C	Specifies the Network Interface Device associated with the calling device if the call is an external incoming call. This parameter is mandatory for all external incoming calls and shall not be provided otherwise.
associatedCalledDevice	AssociatedCalledDeviceID	C	For outgoing external calls, this parameter specifies the Network Interface Device associated with the originally called device. For incoming external calls, this parameter specifies a device within the switching sub-domain associated with the originally called device. This parameter is mandatory for all external outgoing calls and it is optional for external incoming calls.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
establishedConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the establishedConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information attached to the event.

17.2.8.2 Event Causes

Table 17-154 Established—Event Causes

Event Cause	Description	Associated Features
ACD Forward	The call was established at a device (immediately, before being delivered) and is no longer queued at the previous ACD device.	ACD

Table 17-154 Established—Event Causes (continued)

Event Cause	Description	Associated Features
Alternate	The call was reestablished or answered at a device due to the alternate feature.	Alternate Call
Call Back	The call was established at a device (immediately, before being delivered) because of a previously set call back feature.	Call Back Call-Related
Call Forward	The call was established at a device (immediately, before being delivered) after it was forwarded (any type of forwarding).	Call Forwarding
Call Forward—Busy	The call was established at a device (immediately, before being delivered) after it was forwarded because of a busy condition.	Call Forwarding
Call Forward—Immediate	The call was established at a device (immediately, before being delivered) after it was forwarded (forwarding on all conditions).	Call Forwarding
Call Forward—No Answer	The call was established at a device (immediately, before being delivered) after it was forwarded because of a no answer condition.	Call Forwarding
Call Pickup	The call was established at a device via a pickup feature.	Directed Pickup Call, Group Pickup Call
Distributed	The call was distributed, (and was established immediately, before being delivered) to a device because the call has moved from the distribution mechanism (ACD, Hunt) to an available device associated with the distribution mechanism	ACD, Routeing Services
Intrude	The call was established at a device because the Intrude Call service was executed successfully.	Intrude Call
Key Operation	The call was established at a device that has an appearance in a call appearance, bridged, or hybrid device configuration.	Multiple Appearance
Network Signal	The call was established at a device that is outside of the switching sub-domain.	External Calls
New Call	The call was not redirected.	Any Feature
No Available Agents	The call was established immediately, before being delivered, at a device because there were no available devices in a group (ACD)	ACD, Forwarding
Normal	The call was established at a device (a more specific cause cannot be provided).	Any feature
Overflow	The call was established at a device (immediately, before being delivered) after it overflowed a queue, group, or target.	ACD
Override	The call was established at a device as a result of an override (e.g., Single Step Conference Call, Intrude Call) feature.	Intrude Call
Recall	The call was established at a device as part of the recall feature (e.g., due to a time-out associated with a feature that failed to complete) the call was established immediately (e.g. auto answer), before being delivered.	Recall
Redirected	The call was established at a device after it was previously diverted from or deflected to this device the call was established immediately, before being delivered.	Deflect Call, Divert Call

Table 17-154 Established—Event Causes (continued)

Event Cause	Description	Associated Features
Remains in Queue	The call was established at a device and the calling device's position in the ACD queue is retained and the call will be routed to an agent when one becomes available. For example, the call could be established at a VRU or other automated answering equipment while the calling device retains its position in the ACD queue.	ACD
Resources Not Available	The call was established at a device (immediately, before being delivered) because some resources were not available (ACD, forwarding).	ACD, Forwarding
Single Step Conference	The call was established at a device (immediately, before being delivered) due to the Single Step Conference feature.	Single Step Conference
Single Step Transfer	The call was established at a device (immediately, before being delivered) due to the Single Step Transfer feature.	Single Step Transfer
Timeout	The event was generated because a trunk timer expired. It is not generated as the result of a network event or condition.	External Call

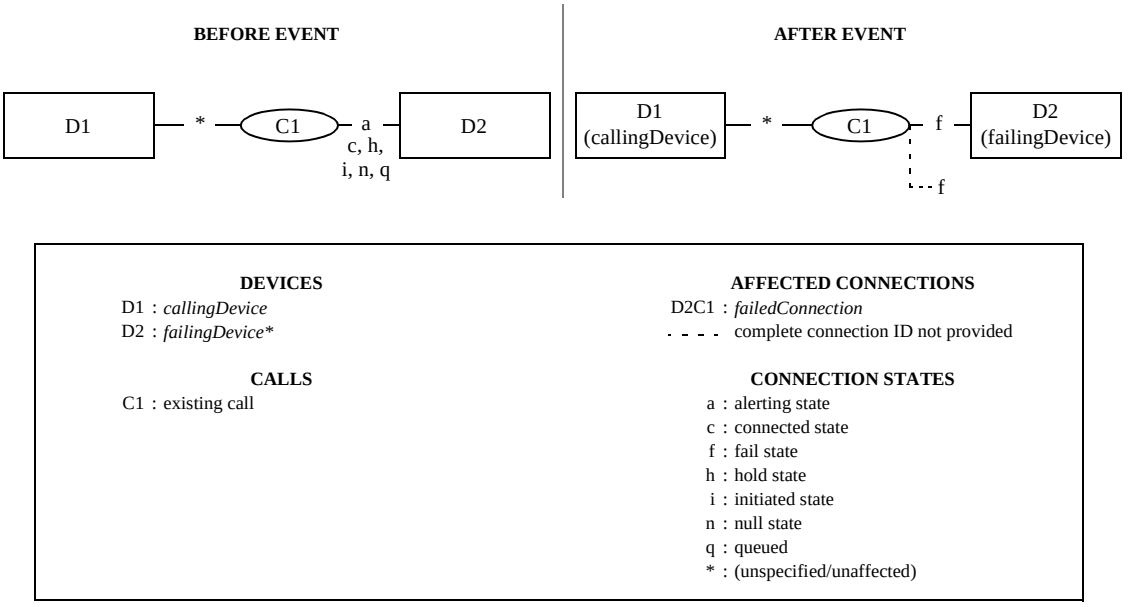
17.2.8.3 Functional Requirements

1. A computing function will not see an Established event after a Service Initiated event (e.g. because of Make Call with prompting) indicating that the calling device is connected in the call. Instead, the computing function will see an Originated event indicating connection into the call.
2. When an appearance from a bridged device configuration connects into a call, an Established event is sent. The event will contain the following type of information:
   - The event cause will be Normal.
   - The establishedConnection will be the connection identifier of the specific appearance.
   - The answeringDevice will depend on the type of referencing model that is supported by the switching function (refer to 6.1.7, "Referencing Devices, Elements, Appearances and Device Configurations", on page 36).
   - The calledDevice or associatedCalledDevice will contain the device identifier associated with the logical element of the device.
3. When observing a call or a device in a call, and the call diverts from a device in the call, if the switching function does not provide the Diverted event for all devices in a call or for call-type monitors (as indicated through the capabilities exchange services option), the computing function shall use the answeringDevice, calledDevice, lastRedirectionDevice, and cause (eventCause parameter type) parameters to properly track the progress of the call as a result of the redirection. See 6.8.6, "Tracking a Diverted Call", on page 57 for more information.
4. If there is more than one calling device (i.e. a conference calling back to a device), then the originatingNIDConnection parameter will not be present.
5. For external incoming calls, the contents of the networkCallingDeviceID and the networkCalledDeviceID parameters shall not change as long as the associatedCallingDevice remains in the call. This differs from the callingDeviceID and the calledDeviceID parameters in that these parameters may change as a result of features such as transfer, forwarding, and conferencing, etc.

17.2.9 Failed

The Failed event indicates that a call cannot be completed or a connection has entered the Fail state for any of the reasons described in the Event Causes table on page 298.

Figure 17-40 Failed Event



Refer to 6.8.2, “Connection Failure”, on page 53 for a complete description of partial connection IDs.

In some situations D2 can be the calling device.

17.2.9.1 Event Parameters

Table 17-155 Failed—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
failedConnection	ConnectionID	M	Specifies the connection that failed. (See Functional Requirement #3 at the end of this section.)
failingDevice	SubjectDeviceID	M	Specifies the device that failed.
callingDevice	CallingDeviceID	M	Specifies the calling device.
calledDevice	CalledDeviceID	M	Specifies the originally called device.
lastRedirectionDevice	RedirectionDeviceID	M	Specifies the previously known redirected-from device.
originatingNIDConnection	ConnectionID	O	Specifies the connection of the Network Interface Device (NID) that the call originated from. If this parameter is supported, it shall be provided if there is an originating NID associated with the call, otherwise it shall not be provided.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the failing device: Fail • For the other devices left in the call: (unaffected) This parameter is mandatory for events generated for device-type monitors and otherwise shall not be provided.

Table 17-155 Failed—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call and shall otherwise not be provided.
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
networkCallingDevice	NetworkCallingDeviceID	O	Specifies the original calling device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
networkCalledDevice	NetworkCalledDeviceID	O	Specifies the original called device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
associatedCallingDevice	AssociatedCallingDeviceID	C	Specifies the Network Interface Device associated with the otherDevice if the call is an external incoming call. This parameter is mandatory for all external incoming calls and shall not be provided otherwise.
associatedCalledDevice	AssociatedCalledDeviceID	C	For outgoing external calls, this parameter specifies the Network Interface Device associated with the originally called device. For incoming external calls, this parameter specifies a device within the switching sub-domain associated with the originally called device. This parameter is mandatory for all external outgoing calls and it is optional for external incoming calls.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
failedConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the failedConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.2.9.2 Event Causes

Table 17-156 Failed—Event Causes

Event Cause	Description	Associated Features
Blocked	The call failed after a device has disconnected from a call leaving one other device remaining in the call.	Connection Failure

Table 17-156 Failed—Event Causes (continued)

Event Cause	Description	Associated Features
Busy	The call failed after it encountered a busy or unavailable device.	Connection Failure
Call Cancelled	The call failed before the associated device has gone on-hook.	Connection Failure
Call Not Answered	The call failed because it was not answered before a timer expired.	Connection Failure
Destination Not Obtainable	The call failed because it could not reach the destination.	Connection Failure
Destination Out of Order	The call failed because it encountered a destination out of service.	Connection Failure
Do Not Disturb	The call failed because it encountered a device that has the do not disturb feature set.	Do Not Disturb, Call Forwarding
Incompatible Destination	The call failed because it encountered an incompatible destination.	Connection Failure
Invalid Account Code	The call failed because of an invalid account code.	Connection Failure
Invalid Number Format	The call failed because the dialled number is incorrect	Connection Failure
Key Operation in Use	The call failed because an appearance is associated with an exclusive bridged device configuration and that the appearance is disabled until the call is terminated or until the call moves away from the device.	Multiple Appearance
Lockout	The call failed because it encountered an inter-digit time-out while dialling.	Connection Failure
Maintenance	The call failed because it encountered a facility or endpoint in a maintenance condition.	Connection Failure
Network Congestion	The call failed because it encountered a congested network. In some circumstances, this event cause indicates that the user is listening to a special signal tone from a network. The tone may be accompanied by a voiced statement similar to “All circuits are busy...”	Connection Failure
Network Not Obtainable	The call failed because it could not reach a destination network.	Connection Failure
Network Out of Order	The call failed because it encountered a network that is out of order.	Connection Failure
Network Signal	The call failed because it encountered a problem after it left the switching sub-domain.	External Calls
Normal	Normal cause for event.	Any feature
Not Available Bearer Service	The call failed because it was requested with a bearer capability that is currently not available.	Connection Failure
Not Supported Bearer Service	The call failed because it was requested with a bearer capability that is currently not supported.	Connection Failure
Number Changed	The call failed because the called number has been changed to a new number and the call cannot be completed.	Connection Failure
Number Unallocated	The call failed because the called number is not allocated to a subscriber.	Connection Failure

Table 17-156 Failed—Event Causes (continued)

Event Cause	Description	Associated Features
Reorder Tone	The call failed because it encountered a reorder condition. When this occurs, the network usually provides Reorder Tone to indicate that a request (call, feature, or supplementary service) was not recognizable. This condition typically results when a user dials a number that is not valid or attempts to obtain a service that is not enabled for that user or device.	Connection Failure
Resources Not Available	The call failed because resources were not available.	Connection Failure
Selected Trunk Busy	The call failed because a specific selected Network Interface Device (e.g., trunk, CO line) is busy.	Connection Failure
Trunks Busy	The call failed because there is no available Network Interface Device (e.g., trunk, CO line).	Connection Failure
Unauthorized Bearer Service	The call failed because it was requested with an unauthorized bearer capability.	Connection Failure

17.2.9.3 Functional Requirements

1. With an exclusive bridged device configuration, a Failed event is generated for each appearance that does not enter the call. Each event will contain the following type of information:
   - The event cause will be Key Operation In Use.
   - The failedConnection will be the connection identifier of the specific appearance.
   - The failingDevice will depend on the type of referencing model that is supported by the switching function (refer to 6.1.7, “Referencing Devices, Elements, Appearances and Device Configurations”, on page 36).
   - The calledDevice or associatedCalledDevice will contain the device identifier associated with the logical element of the device.
2. When observing a call or a device in a call, and the call diverts from a device in the call, if the switching function does not provide the Diverted event for all devices in a call or for call-type monitors (as indicated through the capabilities exchange services option), the computing function shall use the failingDevice, calledDevice, lastRedirectionDevice, and cause (EventCause parameter type) parameters to properly track the progress of the call as a result of the redirection. See 6.8.6, “Tracking a Diverted Call”, on page 57 for more information.
3. The content of the failedConnection parameter, depending on whether or not the Failed event is reported to the failing device’s monitor, will contain one of the following types of connection identifiers (use the capabilities exchange services to determine which approach is supported by a particular switching function):
   a. A complete connection identifier (i.e., an identifier that contains both a device identifier and call identifier). This indicates that a connection is made with the failing device, and that the Failed event will be reported to all active device-type monitors associated with the call, as well as all call-type monitors associated with the call.
   b. A call identifier only connection identifier (i.e., an identifier that contains a valid call identifier and no device identifier). This indicates that a connection is not made with the failing device, and that the Failed event will only be reported to the active device and call-type monitors associated with the devices that were in the call prior to the failure (i.e., if a device-type monitor was on the failing device, then the Failed event is not reported).

c. A complete connection identifier (i.e., an identifier that contains both a device identifier and call identifier), not being reported for monitors on the failing device. This indicates that the Failed event will only be reported to the active device and call-type monitors associated with the devices that were in the call prior to the failure (i.e., if a device-type monitor was on the failing device, then the Failed event is not reported).

4. For external incoming calls, the contents of the networkCallingDeviceID and the networkCalledDeviceID parameters shall not change as long as the associatedCallingDevice remains in the call. This differs from the callingDeviceID and the calledDeviceID parameters in that these parameters may change as a result of features such as transfer, forwarding, and conferencing, etc.

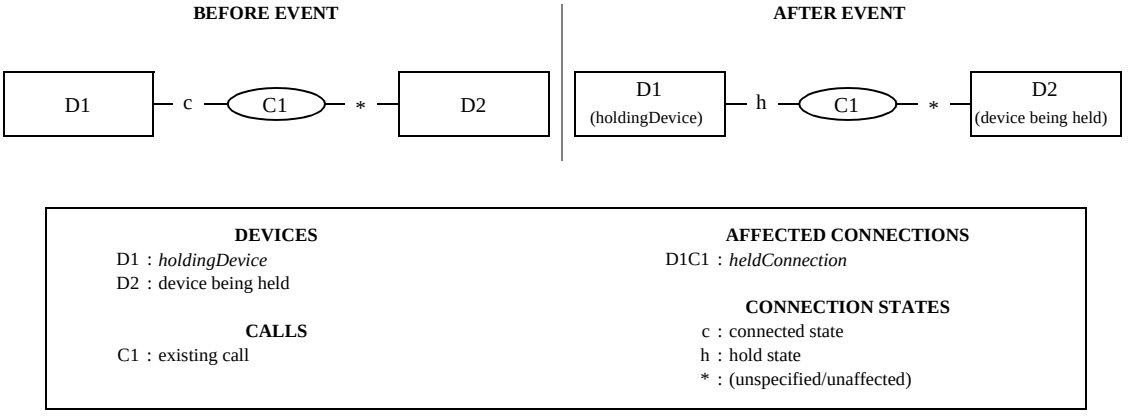
17.2.10 Held

The Held event indicates that a call has been placed on hold.

Common situations that generate this event include:

- Consultation situations (manual and service initiated).
- Hold situations (manual and service initiated).

Figure 17-41 Held Event



17.2.10.1 Event Parameters

Table 17-157 Held—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
heldConnection	ConnectionID	M	Specifies the connection at which the hold was activated.
holdingDevice	SubjectDeviceID	M	Specifies the device at which hold was activated.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the holding device: Hold • For the other devices left in the call: (unaffected) This parameter is mandatory for events generated for device-type monitors.
correlatorData	CorrelatorData	O	Specifies the correlator data associated with the call.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.

Table 17-157 Held—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
heldConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the heldConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies on-standardized information.

17.2.10.2 Event Causes

Table 17-158 Held —Event Causes

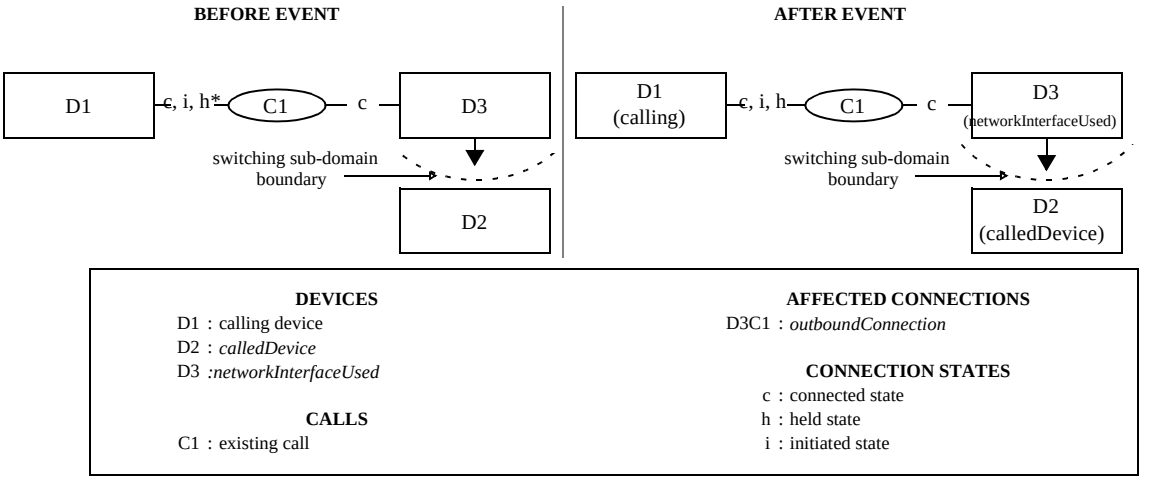
Event Cause	Description	Associated Features
Alternate	The call was held at a device as part of an alternate feature.	Alternate Call
Conference	The call was held at a device as a part of a consultation with the intended purpose of establishing a conference.	Consultation Call with a consultOptions of ConferenceOnly
Consultation	The call was held at a device as part of a consultation.	Consultation Call
Intrude	The call was held at a device because the Intrude Call service was executed successfully.	Intrude Call
Maintenance	The call was placed on hold at a device that entered maintenance conditions (e.g. mobile handset out of coverage)	Maintenance
Network Signal	The call was held by a device that is outside of the switching sub-domain.	External Call
Normal	The call was held at a device (a more specific cause cannot be provided).	Any feature
Recall	The call was held at a device due to the activation of the recall feature.	Recall
Suspend	The call was temporarily placed on hold at a device that went onhook.	Call Clearing
Transfer	The call was held at a device as part of a transfer feature.	Consultation Call with a consultOptions of Transfer Only

17.2.11 Network Capabilities Changed

The Network Capabilities Changed event indicates that a situation occurred during a call's progress in a public or private network that modifies its signalling capability (i.e., inter-networking). It does not indicate a change in the connection state of the Network Interface Device (e.g., trunk, CO Line) through which the call has accessed the network.

This event shall always be preceded by a Network Reached event that included the networkCapability parameter. The event may be repeated according to the situations encountered in the network.

Figure 17-42 Network Capabilities Changed Event



Note that the * held state when an external outgoing call in progress is placed held.

17.2.11.1 Event Parameters

Table 17-159 Network Capabilities Changed—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
outboundConnection	ConnectionID	M	Specifies the outbound connection of the network interface device.
networkInterfaceUsed	SubjectDeviceID	M	Specifies the Network Interface Device (e.g., trunk, CO Line) that was selected.
calledDevice	CalledDeviceID	M	Specifies the destination device.

Table 17-159 Network Capabilities Changed—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
progressIndicator	Structure	M	Specifies an interworking situation encountered in the public or private network outside the switching sub-domain. As a consequence of entering the network, the level of event reporting associated with this call may be reduced. This parameter consists of the following components: 1. progressLocation (M) - It may be one of the following: • User • Private network serving the local user • Public network serving the local user • Transit network • Public network serving the remote user • Private network serving the remote user • Local interface controlled by the signalling link • International Network • Network beyond interworking point • Other 2. progressDescription (M) - It may be one of the following: • ISDN Progress Description - This information is derived from ETSI ETS 300 182: 1993. • QSIG Progress Description - This information is derived from ECMA-143. • Other
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the network interface device: Connected • For the other devices left in the call: (Unaffected) This parameter is mandatory for events generated for device-type monitors.
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call and shall otherwise not be provided.
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
networkCapability	Structure	O	Specifies the type of network reached and the Call Control events supported by the network. It includes the following components: 1. networkType (M) - The complete set of possible values is: • ISDN public • Non-ISDN public • ISDN private • Non-ISDN private • Other 2. eventsProvided (O) - This is a bitmap of all of the Call Control events defined in this Standard.
cause	EventCause	M	Specifies the reason for the event.

Table 17-159 Network Capabilities Changed—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
outboundConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the outboundConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

17.2.11.2 Event Causes

Table 17-160 Network Capabilities Changed—Event Causes

Event Cause	Description	Associated Features
Network Signal	The call encountered an interworking situation.	External Call

17.2.12 Network Reached

The Network Reached event indicates that a call has cut through the switching sub-domain boundary to another network; that is, has reached and engaged a Network Interface Device (e.g., trunk, CO Line). This event indicates that there may be a reduced level of event reporting and possibly no additional device feedback, except connection/ call clearing, provided for this device in the call due to a lack of network signalling. The level of signalling provided by the network may be indicated by the networkCapability parameter.

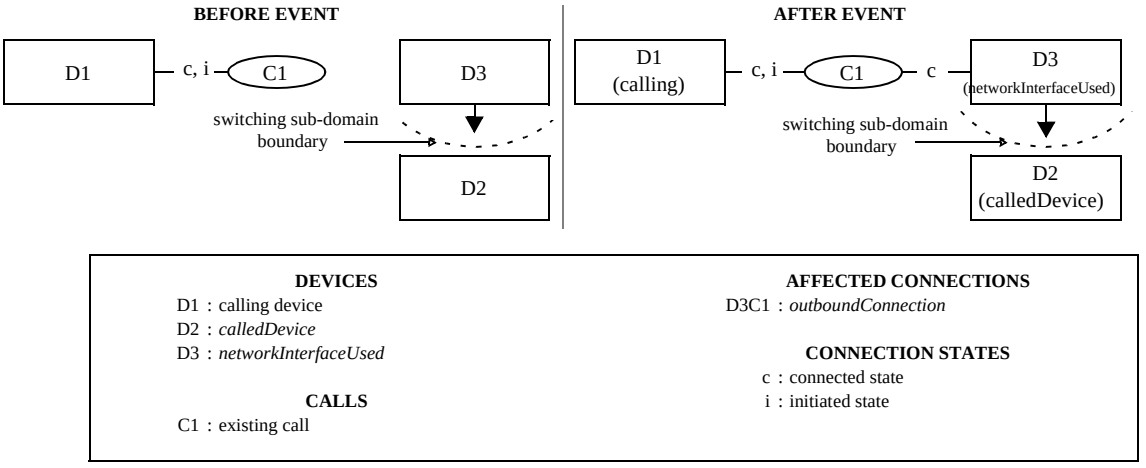
Additionally, the computing function should assume that it cannot directly manipulate the far-end device associated with the Network Interface Device.

This event is never sent for calls made to devices that are within the switching sub-domain. This event indicates that a connection with a Network Interface Device has reached the connected state, and that further events for that connection refer to the state of the endpoint which the Network Interface Device is associated.

A common situation that generates this event includes:

- An outgoing call has cut-through at a network interface device and further call progress information, such as the Delivered and Established events, may not be available.

Figure 17-43 Network Reached Event



17.2.12.1 Event Parameters

Table 17-161 Network Reached—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
outboundConnection	ConnectionID	M	Specifies the outbound connection associated with the call that is leaving the switching sub-domain.
networkInterfaceUsed	SubjectDeviceID	M	Specifies the Network Interface Device that was selected.
callingDevice	CallingDeviceID	M	Specifies the calling device.
calledDevice	CalledDeviceID	M	Specifies the originally called device.
lastRedirectionDevice	RedirectionDeviceID	M	Specifies the previously known redirected from device.

Table 17-161 Network Reached—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
originatingNIDConnection	ConnectionID	O	Specifies the connection of the Network Interface Device (NID) that the call originated from. If this parameter is supported, it shall be provided if there is an originating NID associated with the call, otherwise it shall not be provided.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the Network Interface Device: Connected • For the other devices left in the call: (unaffected) This parameter is mandatory for events generated for device-type monitors.
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call and shall otherwise not be provided.
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
networkCapability	Structure	O	Specifies the type of network reached and the Call Control events supported by the network. It includes the following components: 1. networkType (M) - The complete set of possible values is: • ISDN public • Non-ISDN public • ISDN private • Non-ISDN private • Other 2. eventsProvided (O) - This is a bitmap of the Call Control events defined in this Standard.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
outboundConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the outboundConnection connection. If this parameter is not present, then the connection information is switching function specific.
networkCallingDevice	NetworkCallingDeviceID	O	Specifies the original calling device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls (that, in this case are also outgoing calls).

Table 17-161 Network Reached—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
networkCalledDevice	NetworkCalledDeviceID	O	Specifies the original called device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls (that, in this case are also outgoing calls).
associatedCallingDevice	AssociatedCallingDeviceID	C	Specifies the Network Interface Device associated with the calling device if the call is an external incoming call (that, in this case is also an outgoing). This parameter is mandatory for all external incoming calls and shall not be provided otherwise.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information attached to the event.

17.2.12.2 Event Causes

Table 17-162 Network Reached—Event Causes

Event Cause	Description	Associated Features
ACD Forward	The call left the switching sub-domain because it was forwarded from one ACD device to another ACD device and is no longer present at the previous ACD device.	ACD
Call Forward	The call left the switching sub-domain after it was forwarded (any type of forwarding).	Forwarding
Call Forward—Busy	The call left the switching sub-domain after it was forwarded because of a busy condition.	Forwarding
Call Forward—Immediate	The call left the switching sub-domain after it was forwarded (forwarding on all conditions).	Forwarding
Call Forward—No Answer	The call left the switching sub-domain after it was forwarded because of a no answer condition.	Forwarding
Distributed	The call left the switching sub-domain after it was distributed by an ACD or hunt group.	ACD
No Agents Available	The call left the switching sub-domain after it was diverted from a device because there were no available devices in a group (ACD).	ACD, distribution
Normal	The call left the switching sub-domain (a more specific event cause cannot be provided).	Any feature
Overflow	The call left the switching sub-domain after it overflowed a queue, group, or target.	ACD
Redirected	The call left the switching sub-domain after it was diverted or deflected.	Deflect Call, Divert Call
Resources Not Available	The call left the switching sub-domain after it was diverted from a device because there were no resources available to process it.	ACD, distribution
Transfer	The call was transferred to a destination outside the switching sub-domain.	Transfer

17.2.12.3 Functional Requirements

1. When observing a call or a device in a call, and the call diverts from a device in the call, if the switching function does not provide the Diverted event for all devices in a call or for call-type monitors (as indicated through the capabilities exchange services option), the computing function shall use the

networkInterfaceUsed, calledDevice, lastRedirectionDevice, and cause (EventCause parameter type) parameters to properly track the progress of the call as a result of the redirection. See 6.8.6, “Tracking a Diverted Call”, on page 57 for more information.

2. After a call has cut through the switching sub-domain boundary to another network (Network Reached event generated), all subsequent events reported for the endpoint to which the Network Interface Device (e.g., trunk, CO Line) is associated shall include a cause parameter with a cause of Network Signal or a more specific cause representing network information.

3. For external incoming calls, the contents of the networkCallingDeviceID and the networkCalledDeviceID parameters shall not change as long as the associatedCallingDevice remains in the call. This differs from the callingDeviceID and the calledDeviceID parameters in that these parameters may change as a result of features such as transfer, forwarding, and conferencing, etc.

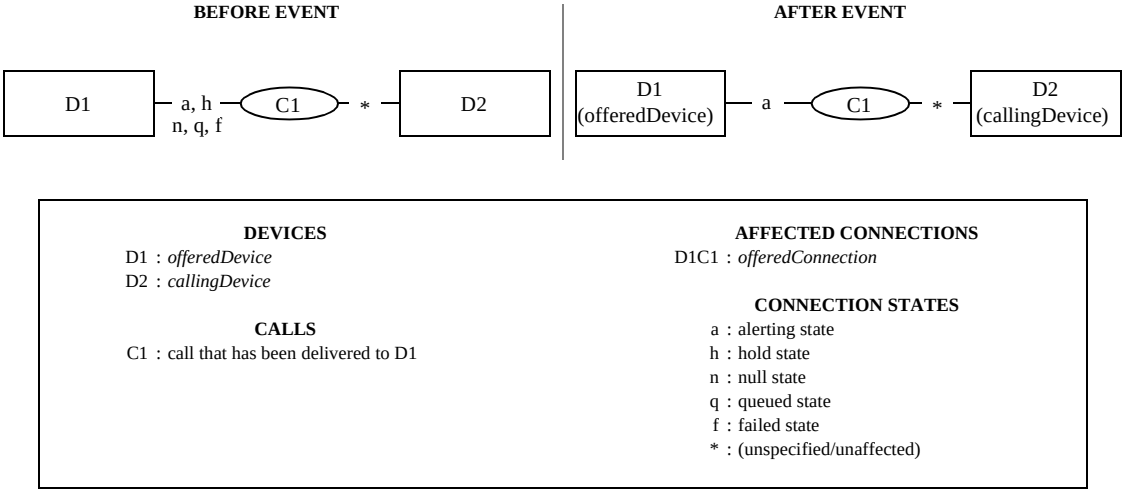
17.2.13 Offered

The Offered event indicates that the connection is in the Offered mode of the alerting state. This indicates that a call is in a pre-delivery state.

In this pre-delivery state, the opportunity exists for a computing function to issue one of a set of supported services (e.g., Accept Call, Clear Connection (“reject”), Deflect Call) or an ISDN device to accept or reject the call. From the calling side perspective, the call is not delivered at the called device. As a consequence, delivery information such as Ringback indication and/or Network signalling is not provided. For example, the device makes no ringing sounds while in the Offered mode of the Alerting state.

The connection may transit to the Ringing mode of the alerting state after the call is accepted (See 17.1.1, “Accept Call”, on page 186) or after a time out period, for example.

Figure 17-44 Offered Event



17.2.13.1 Event Parameters

Table 17-163 Offered—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
offeredConnection	ConnectionID	M	Specifies the connection that is alerting.
offeredDevice	SubjectDeviceID	M	Specifies the device that is alerting.
callingDevice	CallingDeviceID	M	Specifies the calling device.
calledDevice	CalledDeviceID	M	Specifies the originally called device.
lastRedirectionDevice	RedirectionDeviceID	M	Specifies the previously known redirected from device.
originatingNIDConnection	ConnectionID	O	Specifies the connection of the Network Interface Device (NID) that the call originated from. If this parameter is supported, it shall be provided if there is an originating NID associated with the call, otherwise it shall not be provided. See Functional Requirement #3.

Table 17-163 Offered—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the alerting device: Alerting • For the calling device: (unaffected) This parameter is mandatory for events generated for device-type monitors.
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call and shall otherwise not be provided.
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
networkCallingDevice	NetworkCallingDeviceID	O	Specifies the original calling device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
networkCalledDevice	NetworkCalledDeviceID	O	Specifies the original called device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
associatedCallingDevice	AssociatedCallingDeviceID	C	Specifies the Network Interface Device associated with the calling device if the call is an external incoming call. This parameter is mandatory for all external incoming calls and shall not be provided otherwise.
associatedCalledDevice	AssociatedCalledDeviceID	C	For outgoing external calls, this parameter specifies the Network Interface Device associated with the originally called device. For incoming external calls, this parameter specifies a device within the switching sub-domain associated with the originally called device. This parameter is mandatory for all external outgoing calls and it is optional for external incoming calls.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
offeredConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the offeredConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information attached to the event.

17.2.13.2 Event Causes

Table 17-164 Offered—Event Causes

Event Cause	Description	Associated Features
ACD Busy	The call was offered to an ACD device that has currently no available agents.	ACD, Consultation Call
ACD Forward	The call was offered to a new device and is no longer queued at the previous ACD device.	ACD
ACD Saturated	The call was offered to an ACD device that has currently no internal resources available (including agents).	ACD, Consultation Call
Call Back	The call was offered to a device because of a previously set call back feature.	Call Back Call-Related
Call Forward	The call was offered to a device after it was forwarded (any type of forwarding).	Call Forwarding
Call Forward—Busy	The call was offered to a device after it was forwarded because of a busy condition.	Call Forwarding
Call Forward—Immediate	The call was offered to a device after it was forwarded (forwarding on all conditions).	Call Forwarding
Call Forward—No Answer	The call was offered to a device after it was forwarded because of a no answer condition.	Call Forwarding
Distributed	The call was offered to a device because the call has moved from the distribution mechanism (ACD, Hunt) to an available device associated with the distribution mechanism.	ACD, Routeing Services
Entering Distribution	The call was offered to a distribution mechanism (ACD, Hunt) for the purpose of being distributed.	ACD, Routeing Services
Key Operation	The call was offered to a device that has an appearance in a call appearance, bridged, or hybrid device configuration.	Multiple Appearance
Multiple Alerting	The call that is alerting the subject device is also alerting other devices.	Multiple Alerting
Network Signal	The call was offered to a device that is outside of the switching sub-domain.	External Calls
New Call	The call was not redirected.	Any feature
No Available Agents	The call was offered to a device because there were no available devices in a group (ACD).	ACD, Forwarding
Normal	The call was offered to a device (a more specific cause cannot be provided).	Any feature
Overflow	The call was offered to a device after it overflowed a queue, group, or target.	ACD
Override	The call was offered to a device as a result of an override (e.g., Intrude Call) feature.	Intrude Call
Recall	The call was offered to a device as part of the recall feature (e.g., due to a time-out associated with a feature that failed to complete).	Recall
Redirected	The call was offered to a device after it was diverted from or deflected to this device.	Deflect Call, Divert Call
Remains in Queue	The call was offered to a device while the calling device's position in the queue is retained. For example, the call could be offered to a VRU or other automated answering equipment while the calling device retains its position in the ACD queue.	ACD, Queuing, Single Step Conference, Join Call

Table 17-164 Offered—Event Causes (continued)

Event Cause	Description	Associated Features
Resources Not Available	The call was offered because some resources were not available (ACD, forwarding).	ACD, Forwarding
Single Step Transfer	The call alerted the device due to a Single Step Transfer service.	Single Step Transfer
Single Step Conference	The call alerted the device due to a Single Step Conference service.	Single Step Conference
Timeout	The event was generated because a trunk timer expired. It is not generated as the result of a particular network event or condition.	External calls

17.2.13.3 Functional Requirements

1. Switching function implementations vary in their support for call control services while the Offered mode of the alerting state. The computing function shall determine supported services for Offered mode through the capabilities exchange services.
2. The switching function may support an Offered mode time-out. For example, if no service is applied to the connection within an switching function-specific period, it will transit from the Offered mode to the Ringing mode of the alerting state.
3. If there is more than one calling device (i.e. a conference calling back to a device), then the originatingNIDConnection parameter will not be present.
4. When a call is offered to a bridged device configuration, multiple Offered events are sent (one for each appearance of the device configuration). Each event will contain the following type of information:
   - The event for the appearance which is offered the call first will have the appropriate event cause on why the call is being offered to the device configuration. All subsequence events for the other appearances will have an event cause of Key Operation.
   - The offeredConnection will be the connection identifier of the specific appearance.
   - The offeredDevice will depend on the type of referencing model that is supported by the switching function (refer to 6.1.7, “Referencing Devices, Elements, Appearances and Device Configurations”, on page 36).
   - The calledDevice or associatedCalledDevice will contain the device identifier associated with the logical element of the device.
5. When observing a call or a device in a call, and the call diverts from a device in the call, if the switching function does not provide the Diverted event for all devices in a call or for call-type monitors (as indicated through the capabilities exchange services option), the computing function shall use the offeredDevice, calledDevice, lastRedirectionDevice, and cause (eventCause parameter type) parameters to properly track the progress of the call as a result of the redirection. See 6.8.6, “Tracking a Diverted Call”, on page 57 for more information.
6. For external incoming calls, the contents of the networkCallingDeviceID and the networkCalledDeviceID parameters shall not change as long as the associatedCallingDevice remains in the call. This differs from the callingDeviceID and the calledDeviceID parameters in that these parameters may change as a result of features such as transfer, forwarding, and conferencing, etc.

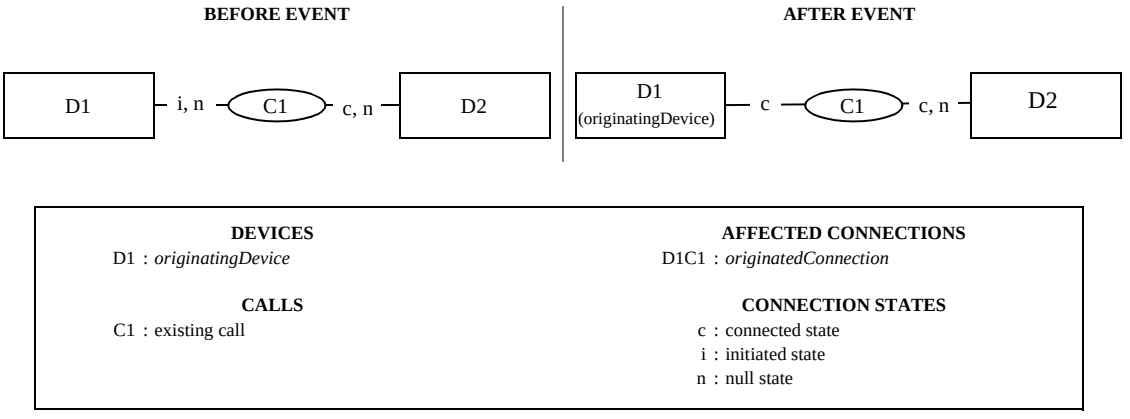
17.2.14 Originated

The Originated event indicates that a call is being attempted from a device. It implies that input activity for the call is complete and that a call (rather than a feature) has been requested.

Common situations that generate this event when the switch has originated a call at the originating device are:

- Due to the execution of the Make Call service.
- Due to the execution of the Consultation Call service.
- After a user has completed manually dialling a number.
- When an external incoming call originates from a Network Interface Device (e.g., trunk, CO line)

Figure 17-45 Originated Event



Note that the connected state for D2C1 is when D2 is a NID.

17.2.14.1 Event Parameters

Table 17-165 Originated—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
originatedConnection	ConnectionID	M	Specifies the connection at which the call originated.
callingDevice	SubjectDeviceID	M	Specifies the calling device.
calledDevice	CalledDeviceID	M	Specifies the originally called device.
originatingDevice	DeviceID	O	Specifies the originating device. The originating device represents a device that originated a call in the switching sub-domain on behalf of the calling device (e.g., a group of stations). The originating device shall not represent a Network Interface Device. This parameter may be provided for external outgoing and internal calls when the originating device is different from the calling device. This parameter shall not be provided for external incoming calls or when the originating device is identical to the calling device.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the device initiating the call: Connected This parameter is mandatory for events generated for device-type monitors.

Table 17-165 Originated—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call and shall otherwise not be provided.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
networkCallingDevice	NetworkCallingDeviceID	O	Specifies the original calling device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
networkCalledDevice	NetworkCalledDeviceID	O	Specifies the original called device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
associatedCallingDevice	AssociatedCallingDeviceID	C	Specifies the Network Interface Device associated with the calling device if the call is an external incoming call. This parameter is mandatory for all external incoming calls and shall not be provided otherwise.
associatedCalledDevice	AssociatedCalledDeviceID	C	For outgoing external calls, this parameter specifies the Network Interface Device associated with the originally called device. For incoming external calls, this parameter specifies a device within the switching sub-domain associated with the originally called device. This parameter is mandatory for all external outgoing calls and it is optional for external incoming calls.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
originatedConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the originatedConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information attached to the event.

17.2.14.2 Event Causes

Table 17-166 Originated—Event Causes

Event Cause	Description	Associated Features
Call Back	A call was originated from a device because a previously set call back feature.	Call Back Call-Related, Call Back Non-Call-Related
Conference	A call was originated from a device as part of consultation with the intended purpose of establishing a conference.	Consultation Call with a consultOptions of ConferenceOnly

Table 17-166 Originated—Event Causes (continued)

Event Cause	Description	Associated Features
Consultation	A call was originated from a device as part of a consultation call.	Consultation Call
Make Call	A call was originated from a device as the result of the Make Call service.	Make Call (without prompting)
New Call	A call was originated.	Any feature
Normal	A call was originated from a device (a more specific event cause cannot be provided).	Any feature, Make Call
Transfer	A call was originated from a device as part of a consultation for the purpose of transferring a call.	Consultation Call with a consultOptions of TransferOnly

17.2.14.3 Functional Requirements

1. This event will only be generated when the computing function has a device-type or call-type monitoring applied to a device that is initiating a call.
2. This event will only be generated on an incoming call from a device outside the switching sub-domain when the device is a Network Interface Device that can have monitoring applied to it.
3. For external incoming calls, the contents of the networkCallingDeviceID and the networkCalledDeviceID parameters shall not change as long as the associatedCallingDevice remains in the call. This differs from the callingDeviceID and the calledDeviceID parameters in that these parameters may change as a result of features such as transfer, forwarding, and conferencing, etc.

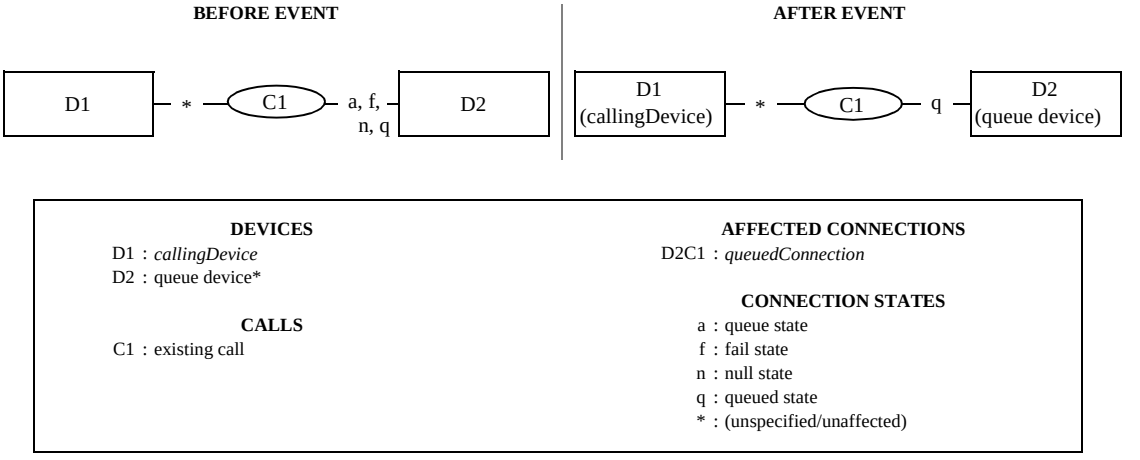
17.2.15 Queued

The Queued event indicates that a call has been queued.

Common situations that generate this event include:

- A call is queued at an ACD or hunt group device.
- A call is queued (camped on or parked, for example) at a device.

Figure 17-46 Queued Event



*There are some situations where D2 can be the calling device.

17.2.15.1 Event Parameters

Table 17-167 Queued—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
queuedConnection	ConnectionID	M	Specifies the queued connection.
queue	SubjectDeviceID	M	Specifies the queued device.
callingDevice	CallingDeviceID	M	Specifies the calling device.
calledDevice	CalledDeviceID	M	Specifies the originally called device.
lastRedirectionDevice	RedirectionDeviceID	M	Specifies the previously known redirected from device.
numberQueued	Value	O	Specifies the total number of calls in queue, including this call.
callsInFront	Value	O	Specifies the number of calls ahead of the call when it was enqueued (this call is not counted in this number).
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the queueing device: Queued • For the other devices left in the call: (unaffected) This parameter is mandatory for events generated for device-type monitors.
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call and shall otherwise not be provided.

Table 17-167 Queued—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
networkCallingDevice	NetworkCallingDeviceID	O	Specifies the original calling device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
networkCalledDevice	NetworkCalledDeviceID	O	Specifies the original called device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
associatedCallingDevice	AssociatedCallingDeviceID	C	Specifies the Network Interface Device associated with the calling device if the call is an external incoming call. This parameter is mandatory for all external incoming calls and shall not be provided otherwise.
associatedCalledDevice	AssociatedCalledDeviceID	C	For outgoing external calls, this parameter specifies the Network Interface Device associated with the originally called device. For incoming external calls, this parameter specifies a device within the switching sub-domain associated with the originally called device. This parameter is mandatory for all external outgoing calls and it is optional for external incoming calls.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
queuedConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the queuedConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specified non-standardized information.

17.2.15.2 Event Causes

Table 17-168 Queued—Event Causes

Event Cause	Description	Associated Features
Busy	The call was queued at a called device because the device was active on another call and unable to accept this call.	Make Call, Consultation Call, Deflect Call, Single-Step Conference Call, Single-Step Transfer Call
Call Forward	The call was immediately queued to a device after it was forwarded.	Call Forwarding

Table 17-168 Queued—Event Causes (continued)

Event Cause	Description	Associated Features
Call Forward—Busy	The call was immediately queued to a device after it was forwarded because of a busy condition.	Call Forwarding
Call Forward—Immediate	The call was immediately queued to a device after it was forwarded (forwarding on all conditions).	Call Forwarding
Call Forward—No Answer	The call was immediately queued to a device after it was forwarded because of a no answer condition.	Call Forwarding
Camp On	The call was queued to a device because of the camp on feature.	Camp On Call
Camp On Trunks	The call was queued to a trunk because of the camp on feature.	External Call, Camp On Call
Distribution Delay	The distribution of the call was delayed at a distribution device for distribution device specific reasons (e.g. preconnect announcement).	ACD, queueing
Do Not Disturb	The call was queued at a device after it encountered a device with Do Not Disturb feature set.	Forwarding
Multiple Queueing	The call that is queued is also queued at other devices.	Multiple Queueing
Network Congestion	The call was queued because the call encountered a congested network.	External Call
Network Not Obtainable	The call could not reach a destination network.	External Call
Network Signal	The call was queued to a device outside of the switching sub-domain.	External Call
No Available Agents	The call was queued to a device because there were no available agents.	ACD
Normal	The call was queued to a device (a more specific cause cannot be provided).	Any feature
Overflow	The call was queued to a device after it overflowed a queue, group, or target.	ACD, queueing
Park	The call was queued because of the park feature.	Park Call
Recall	The call was queued to a device as part of the recall feature (e.g., due to a time-out associated with a feature that failed to complete).	Recall
Redirected	The call was queued to a device after it was diverted from or deflected to this device.	Deflect Call, Divert Call
Remains in Queue	The call was queued to a device while the calling device's position in the queue is retained. For example, the call could be queued to a VRU or other automated answering equipment while the calling device retains its position in the ACD queue.	ACD, Queuing, Single Step Conference, Join Call
Resources Not Available	The call was queued to a device because resources were not available.	ACD
Trunks Busy	The call was queued to a device because there is no available Network Interface Device (e.g., Trunk, CO line).	External Call

17.2.15.3 Functional Requirements

1. When observing a call or a device in a call, and the call diverts from a device in the call, if the switching function does not provide the Diverted event for all devices in a call or for call-type monitors (as indicated through the capabilities exchange services option), the computing function shall use the queue, calledDevice, lastRedirectionDevice, and cause (eventCause parameter type) parameters to properly track the progress of the call as a result of the redirection. See 6.8.6, "Tracking a Diverted Call", on page 57 for more information.

2. For external incoming calls, the contents of the networkCallingDeviceID and the networkCalledDeviceID parameters shall not change as long as the associatedCallingDevice remains in the call. This differs from the callingDeviceID and the calledDeviceID parameters in that these parameters may change as a result of features such as transfer, forwarding, and conferencing, etc.

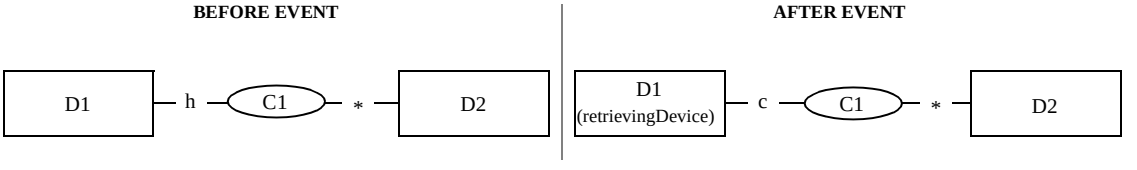
17.2.16 Retrieved

The Retrieved event indicates that a previously held call has been retrieved.

Common situations that generate this event include:

- When a held call is retrieved through the phone using features such as Retrieve, Alternate, etc.
- When a held call is retrieved during the successful execution of the Alternate Call, Reconnect Call, or the Retrieve Call service.

Figure 17-47 Retrieved Event



DEVICES
D1 : *retrievingDevice*
D2 : device in call with D1

CALLS
C1 : existing call

AFFECTED CONNECTIONS
D1C1 : *retrievedConnection*

CONNECTION STATES
c : connected state
h : hold state
* : (unspecified/unaffected)

17.2.16.1 Event Parameters

Table 17-169 Retrieved—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
retrievedConnection	ConnectionID	M	Specifies the connection at which hold was deactivated.
retrievingDevice	SubjectDeviceID	M	Specifies the device at which hold was deactivated.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the retrieving device: Connected • For the other devices left in the call: (unaffected) This parameter is mandatory for events generated for device-type monitors.
correlatorData	CorrelatorData	O	Specifies the correlator data associated with the call.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.

Table 17-169 Retrieved—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
retrievedConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the retrievedConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information attached to the event.

17.2.16.2 Event Causes

Table 17-170 Retrieved—Event Causes

Event Cause	Description	Associated Features
Alternate	The call was retrieved at a device as part of the alternate feature.	Alternate Call
Network Signal	The call was retrieved at a device that is outside of the switching sub-domain.	External Call
Normal	The call was retrieved at a device (a more specific cause cannot be provided).	Any feature, Alternate Call, Reconnect Call, Retrieve Call
Recall	The call was retrieved at a device because a hold timeout expired.	Recall

17.2.16.3 Functional Requirements

1. This event is only generated for the Alternate feature when the device is alternating between a connection that is in the Connected state and a connection that is in the Hold state. Note that this event shall not be generated when the Alternate feature is used to answer an alerting call (the Established event is generated in this case). Refer to 17.1.2, “Alternate Call”, on page 188 for more information.

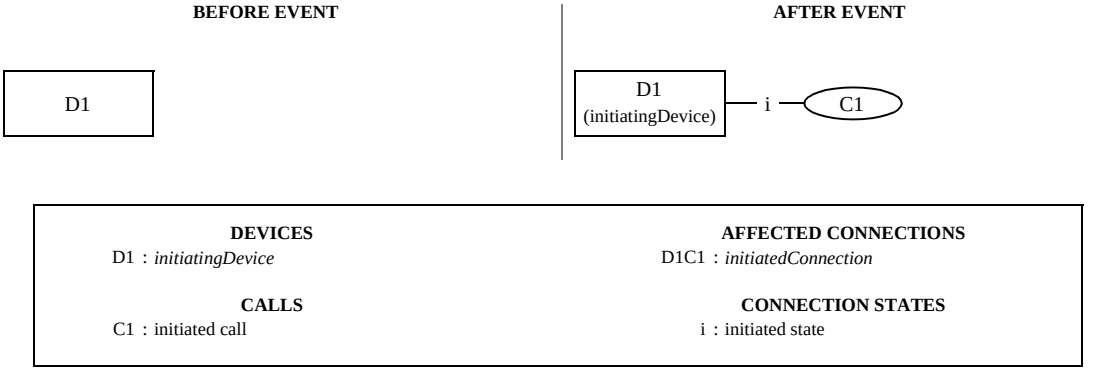
17.2.17 Service Initiated

The Service Initiated event indicates that a telephony service has been initiated at a monitored device. The switching function typically generates this event when “dial-tone” is being provided. This event indicates that either a call may be originated or a feature may be invoked. This event also may indicate that a device is prompting a user.

Common situations that generate this event include:

- When a service or feature prompts a user to take a phone off-hook and that phone is not able to do so without manual intervention.
- A Make Call or Consultation Call service has been invoked and the originating device is initiating a new call that is associated with the originating device.
- When manually invoking any feature at a device for an existing call that requires a new call to be created to input the feature.
- When a device is taken off-hook manually.
- When an incoming call arrives on a monitored Network Interface Device (e.g. trunk, CO line).

Figure 17-48 Service Initiated Event



17.2.17.1 Event Parameters

Table 17-171 Service Initiated—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
initiatedConnection	ConnectionID	M	Specifies the connection at which service was initiated.
initiatingDevice	SubjectDeviceID	M	Specifies the initiating device.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • When the device is initiating a service of some form: Initiated This parameter is mandatory for events generated for device-type monitors.
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call.
cause	EventCause	M	Specifies the reason for the event.

Table 17-171 Service Initiated—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
initiatedConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the initiatedConnection connection. If this parameter is not present, then the connection information is switching function specific.
networkCallingDevice	NetworkCallingDeviceID	O	Specifies the original calling device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
networkCalledDevice	NetworkCalledDeviceID	O	Specifies the original called device information provided by the network for external incoming calls. This parameter is provided only for external incoming calls.
associatedCallingDevice	AssociatedCallingDeviceID	C	Specifies the Network Interface Device associated with the calling device if the call is an external incoming call. This parameter is mandatory for all external incoming calls and shall not be provided otherwise.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information attached to the event.

17.2.17.2 Event Causes

Table 17-172 Service Initiated—Event Causes

Event Cause	Description	Associated Features
Active Participation	The telephony service was initiated at a device and the device is being prompted to go off-hook.	Join Call with a participationType of Active (prompting)
Call Back	The telephony service was initiated at a device and the device is being prompted to go off-hook because of a previously set call back feature.	Call Back Call-Related, Call Back Non-Call-Related
Call Pickup	The telephony service was initiated at a device and the device is being prompted to go off-hook.	Directed Pickup, Group Pickup (prompting)
Conference	The telephony service was initiated at a device as part of a consultation with the intended purpose of establishing a conference.	Consultation Call with a consultOptions of ConferenceOnly
Consultation	The telephony service was initiated at a device as part of a consultation.	Consultation Call
Join Call	The telephony service was initiated at a device and the device is being prompted to go off-hook	Join Call (prompting)
Make Call	The telephony service was initiated at a device and the device is being prompted to go off-hook.	Make Call (prompting)

Table 17-172 Service Initiated—Event Causes (continued)

Event Cause	Description	Associated Features
Make Predictive call	The telephony service was initiated at a (calling) device prior to call activity at a called device as part of a predictive call feature.	Make Predictive Call
New Call	The telephony service was initiated for establishing a connection with another device.	Any feature, Make Call
Normal	A more specific cause cannot be provided.	Any feature
Reserved	A device was reserved prior to call delivery.	Make Predictive Call, incoming call
Silent Participation	The telephony service was initiated at a device and the device is being prompted to go off-hook.	Join Call with a participationType of Silent (prompting)
Transfer	The telephony service was initiated at a device as part of a consultation for the purpose of transferring a call.	Consultation Call with a consultOptions of TransferOnly

17.2.17.3 Functional Requirements

1. Some CSTA services (Make Call, Call Back, Pickup, Join Call) may require to prompt the user of the targeted device in order to take that device off-hook. In this case a Service Initiated event is generated containing the appropriate cause code (Make Call, Call Back, Call Pickup, Join Call). The implementation of this prompting mechanism is switching function specific (display flashing, ring pattern, lamp blinking, etc.).
2. It is switching function dependent as to whether this event is provided to the computing function. For those switching functions that cannot detect prompting, or a device going off-hook and being provided dialtone, the first event seen by the computing function, for the calling device, will be the Originated event.
3. When prompting a call appearance, bridged, or hybrid device configuration as a result of a Make Call service request, only one of the appearances (initiatedConnection) will be prompted.
4. When prompting a call appearance, bridged, or hybrid device configuration as a result of a Call Back service request or Recall feature, only the appearance (initiatedConnection) that issued the Call Back service request or that was the Recall device, will be prompted.
5. For an external incoming call, this event is generated at the Network Interface Device (when this device can have monitoring applied to it).
6. It is not required to send the Service Initiated event for functional (e.g. en-bloc BRI) terminals nor is it required to be sent for calls that are set up without receiving dial tone or other prompting, like CSTA calls initiated with the Make Call service from a hands-free telephone.

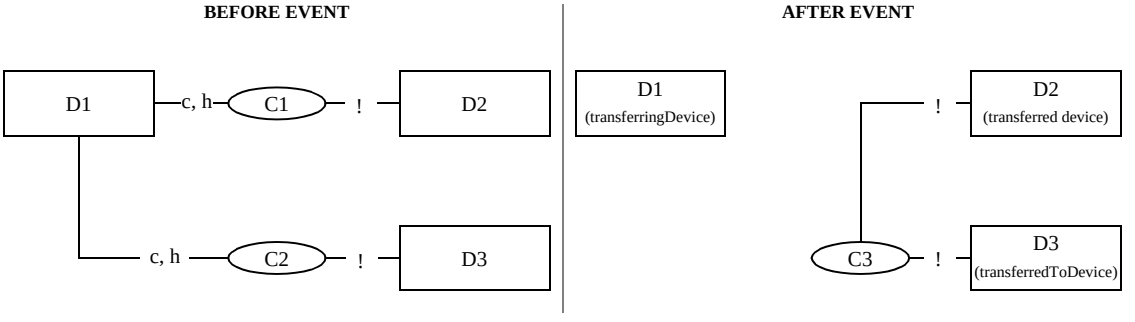
17.2.18 Transferred

The Transferred event indicates that an existing call has been transferred to another device and the transferring device has been dropped from the call. The transferring device does not appear in any future events for the call.

Common situations that generate this event include:

- Two step transferring situations (manual and service initiated).
- Single step transferring situations (manual and service initiated).

Figure 17-49 Transferred Event (Case A: Two Step Transfer)



DEVICES
D1 : *transferringDevice*
D2 : transferred device
D3 : *transferredToDevice*

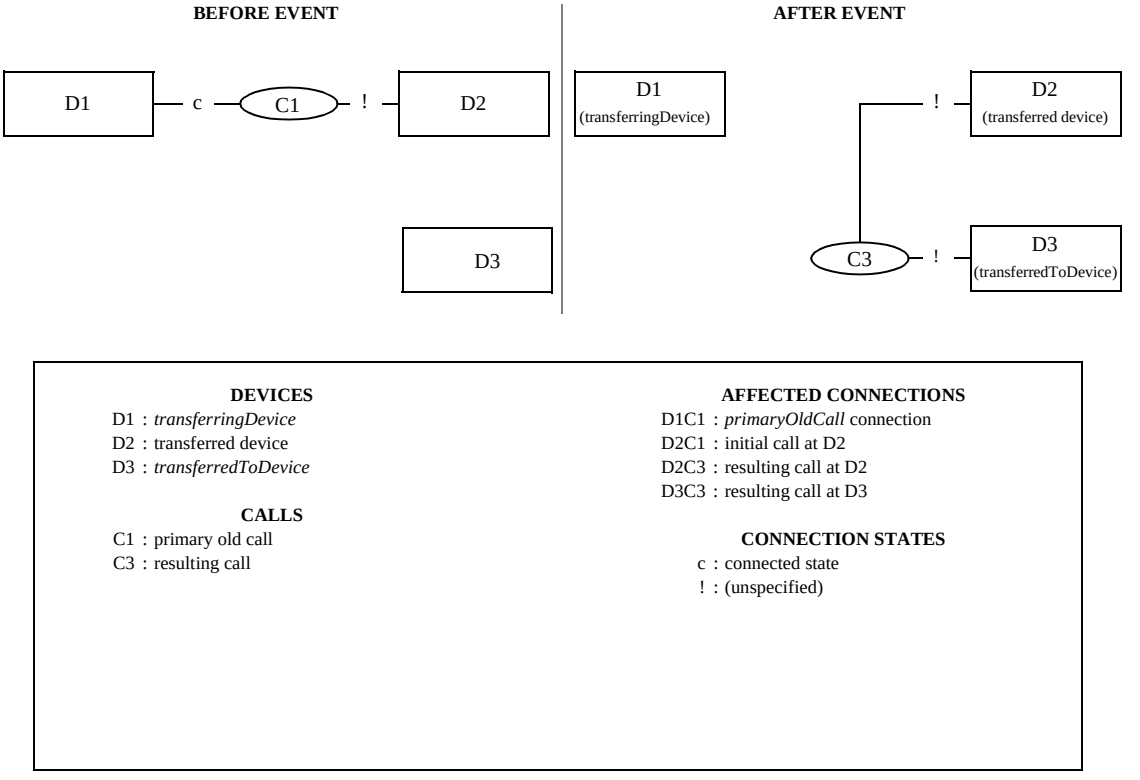
CALLS
C1 : primary old call*
C2 : secondary old call*
C3 : resulting call

AFFECTED CONNECTIONS
D1C1 : *primaryOldCall* connection*
D1C2 : *secondaryOldCall* connection
D2C1 : initial call at D2
D3C2 : initial call at D3
D2C3 : resulting call at D2
D3C3 : resulting call at D3

CONNECTION STATES
c : connected state
h : hold state
! : (unspecified)

* the primary/secondary old call and the primary/secondary old call connections mentioned in this figure are from the perspective of the transferringDevice (D1). See Functional Requirement #1.

Figure 17-50 Transferred Event (Case B: Single Step Transfer)



17.2.18.1 Event Parameters

Table 17-173 Transferred—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
primaryOldCall	ConnectionID	M	Specifies the connection of the primary call. See Functional Requirement #1.
secondaryOldCall	ConnectionID	C	Specifies the connection of the secondary call. If the switching function supports the “fixed-view” option (as indicated by the capability exchange services), this parameter is mandatory. If the switching function supports the “local-view” option, this parameter is mandatory if there are two known calls involved with the transfer (before the transfer is created) from the perspective of the monitored device, otherwise it shall not be provided. See Functional Requirement #1.
transferringDevice	SubjectDeviceID [1]	M	Specifies the device that transferred the call.
transferredToDevice	SubjectDeviceID [1]	M	Specifies the transferred to device.
transferredConnections	ConnectionList	M	Specifies information on each device/ConnectionID in the resulting transferred call that are known by the switching function.

Table 17-173 Transferred—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. The following are the states of the devices in the call. • For the transferring device (any Connection IDs associated with the transfer, i.e., this event should be used for both single and multi-step transfers.): Null • For the other devices associated with the transfer: (unaffected) This parameter is mandatory for events generated for device-type monitors.
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call and shall otherwise not be provided.
userData	UserData	C	Specifies user information that is related to the call. This parameter is mandatory if user data is sent and the parameter is supported, otherwise it shall not be included.
chargingInfo	ChargingInfo	O	Specifies a total value of charging or currency units for the device that transferred the call.
cause	EventCause	M	Specifies the reason for the event.
servicesPermitted	ServicesPermitted	C	Specifies a list of the call control services that can be applied to the local connection. This parameter is mandatory if the switching function supports the Dynamic Feature Availability option (as indicated through the capabilities exchange services), otherwise this parameter is optional.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4 for the complete set of possible values.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information attached to the event.

1. Note that SubjectDeviceID refers to a parameter type—not the subject device of the Transferred event. This parameter type is used to represent the two devices in this event because the two devices are affected by the generation of this event (i.e., the transferring device and the transferred device). However, there is only one subject device of the event and that is the transferring device. For more details on the SubjectDeviceID parameter type, see 12.3.29, “SubjectDeviceID”, on page 117.

17.2.18.2 Event Causes

Table 17-174 Transferred—Event Causes

Event Cause	Description	Associated Features
Network Signal	The call was involved in a transfer located outside of the switching sub-domain.	External Call
Normal	The call was transferred (a more specific event cause cannot be provided).	Transfer
Single Step Transfer	The call was transferred because of a single step transfer.	Single Step Transfer Call
Transfer	The call was transferred because of a two step transfer.	Transfer Call, Two Step Transfer

17.2.18.3 Functional Requirements

1. The contents of the primaryOldCall and the secondaryOldCall parameters may contain either a “fixed view” or a “local view” of the connections at a device before the transfer has been completed. The switching function indicates which view it provides via the connectionView parameter in the capability exchange services.
    - fixed view - for each transferred event generated by monitors placed on different devices in a call, the switching function provides the same information in the primaryOldCall and the secondaryOldCall parameters independent of the monitorType (call or device-type monitor) and independent of the role of the device in the conference (conferencingDevice, addedParty, etc.). The meaning of these parameters for the fixed-view are:
        - primaryOldCall - specifies the first call visible at the transferringDevice.
        - secondaryOldCall - specifies the second call visible at the transferringDevice.
    - local view - for each transferred event generated by monitors placed on different devices in a call, the switching function provides different information in the primaryOldCall and the secondaryOldCall parameters that depends upon which call was made visible first, from the perspective of the monitored device. The meaning of these parameters for the local-view are:
        - primaryOldCall - specifies the first call visible at the monitored device. For example, for a device-type monitor placed on the transferringDevice (two step transfer), this is the first call placed on hold (C1). For a device-type monitor placed on the transferredTo device, this is the first (and only) call involved in call C2 from the perspective of the monitored device.
        - secondaryOldCall - specifies the second call visible at the monitored device. For example, for a device-type monitor placed on the transferringDevice (two step transfer), this is call C2. For a monitor placed on the transferredTo device, there is no secondaryOldCall parameter.
2. The transferredConnections parameter is a list that contains the new ConnectionID, old ConnectionID, DeviceID (values such as ANI, etc.), and for externally located devices the associated Network Interface DeviceID. Refer to 12.2.8, “ConnectionList”, on page 86, for a description of which components are optional and mandatory.
3. The computing function should never assume the reuse of callIDs, although some switching functions may reuse one or the other.

18 Call Associated Features

This clause describes call associated (non-Call Control related) features including:

- Call Associated Feature services
- Call Associated Feature events

Refer to 6.1.4, “Call”, on page 27 and 6.1.5, “Connection”, on page 31 for information on the identifiers used to reference calls and connections.

18.1 Services

Table 18-1 Call Associated Feature Services Summary

Call Associated Feature Service	Description	Pg.
18.1.1 Associate Data	Associates information (such as correlator data, account code, authorisation code, call qualifying data, etc.) with a specified call.	332
18.1.2 Cancel Telephony Tones	Cancels a telephony tone that is being generated on a specified connection.	334
18.1.3 Generate Digits	Generates DTMF tones or rotary pulses on behalf of a connection in a call.	336
18.1.4 Generate Telephony Tones	Generates a specified telephony tone on behalf of a connection in a call.	338
18.1.5 Send User Information	Sends user-to-user information from a specified connection in a call.	341

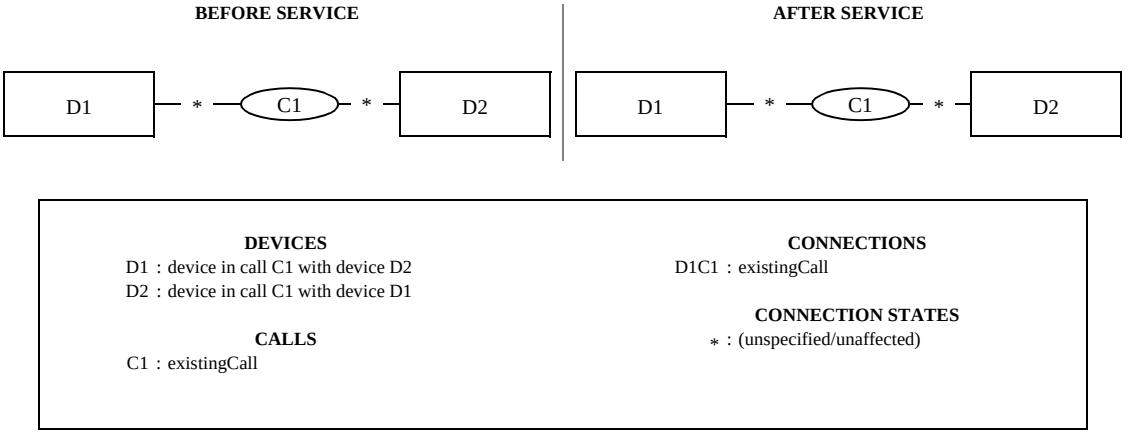
18.1.1 Associate Data

C → S

The Associate Data service associates computing function information (such as correlator data, account code, authorization code, call qualifying data, etc.) with a specified call.

This service does not affect the state or progress of a call.

Figure 18-1 Associate Data Service



Note that the existingCall connectionID could either be the D1C1 or the D2C1 connections in Figure 18-1.

18.1.1.1 Service Request

Table 18-2 Associate Data—Service Request

Parameter Name	Type	M/ O/C	Description
existingCall	ConnectionID	M	Specifies the call with which the information is to be associated.
accountCode [1]	AccountInfo	C	Specifies the account code to associate with the call.
authCode [1]	AuthCode	C	Specifies the authorization code to allow the call.
correlatorData [1]	CorrelatorData	C	Specifies the correlator data to associate with the call.
callQualifyingData [1]	CallQualifyingData	C	Specifies the call qualifying data (wrap code, walk away code, etc.) to associate with the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

1. At least one of these parameters shall be specified on this service (see Functional Requirement #1).

18.1.1.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

18.1.1.2.1 Positive Acknowledgement

Table 18-3 Associate Data—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

18.1.1.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

18.1.1.3 Operational Model

18.1.1.3.1 Connection State Transitions

There are no connection state changes due to this service.

18.1.1.3.2 Device-Type Monitoring Event Sequences

Table 18-4 Associate Data—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event
D1	D1C1	Call Information
D2	D2C1	Call Information

18.1.1.3.3 Call-Type Monitoring Event Sequences

Table 18-5 Associate Data—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event
C1	D1C1/D2C1	Call Information

18.1.1.3.4 Functional Requirements

1. At least one of the service specific parameters (accountCode, authCode, correlatorData, callQualifyingData) shall be provided in the service request, otherwise the service shall be rejected.
2. The Call Information event is generated as the result of this service.
3. For a complete description of the behaviour of correlator data, refer to 6.1.4.3, “Correlator Data”, on page 28.
4. When either the accountCode or callQualifyingData parameter is provided on the service request, the callID component of the connectionID in the service request may be omitted or may refer to the connectionID that no longer exists at the device. This may occur when a user enters a wrap code for a call that has just cleared, for example. It is switching function specific (as indicated by the capability exchange services) whether the switching function rejects the service (with an invalid connectionID) or accepts the service under this condition.
5. When accountInfo, authorisationCode, or callQualifyingData is provided, the ConnectionID in the service request shall be related to the device on whose behalf the data is being associated with.

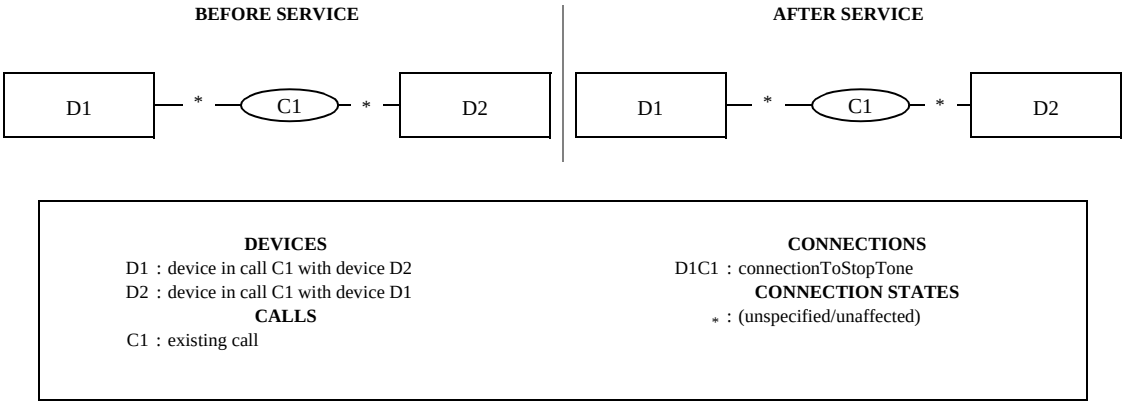
18.1.2 Cancel Telephony Tones

C → S

The Cancel Telephony Tones service cancels a telephony tone that is being sent on a connection by the Generate Telephony Tones service.

This service does not affect the state or progress of a call.

Figure 18-2 Cancel Telephony Tones Service



18.1.2.1 Service Request

Table 18-6 Cancel Telephony Tones—Service Request

Parameter Name	Type	M/ O/C	Description
connectionToStopTone	ConnectionID	M	Connection on which the tones are being generated.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

18.1.2.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

18.1.2.2.1 Positive Acknowledgement

Table 18-7 Cancel Telephony Tones—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

18.1.2.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

18.1.2.3 Operational Model

18.1.2.3.1 Connection State Transitions

There are no connection state changes due to this service.

18.1.2.3.2 Device-Type Monitoring Event Sequences

Table 18-8 Cancel Telephony Tones—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event
D1 (device in call C1 with device D2)	D1C1 (connectionToStopTone)	Telephony Tones Generated (with the toneGenerated parameter not provided)

18.1.2.3.3 Call-Type Monitoring Event Sequences

Table 18-9 Cancel Telephony Tones—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event
C1 (existingCall)	D1C1 (connectionToStopTone)	Telephony Tones Generated (with the toneGenerated parameter not provided)

18.1.2.3.4 Functional Requirements

None.

18.1.3 Generate Digits

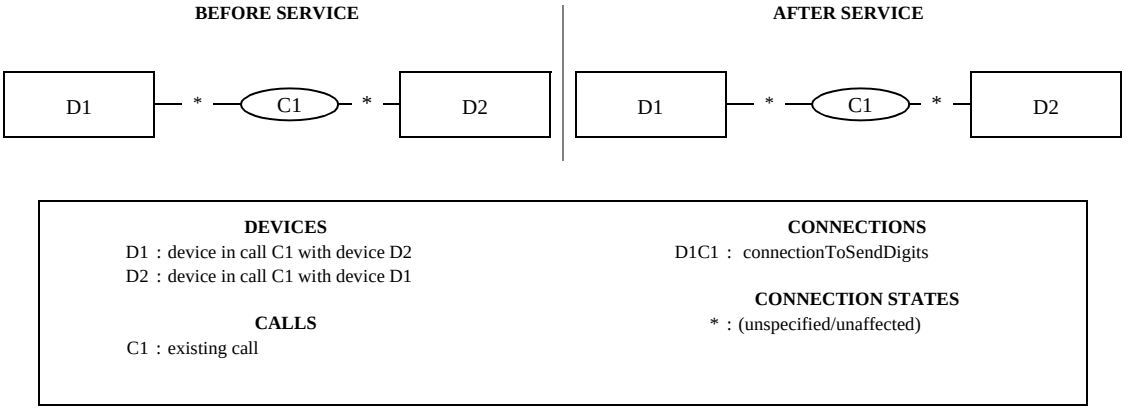
C → S

The Generate Digits service causes a series of digits to be sent on behalf of a connection in a call. The digits may be sent in the form of DTMF tones or rotary pulses. This service also supports optional parameters to control digit generation.

This service is used for generating end-to-end information that is to be sent to a device in a call (i.e., not to address/ select a device).

This service does not affect the state or progress of a call.

Figure 18-3 Generate Digits Service



18.1.3.1 Service Request

Table 18-10 Generate Digits—Service Request

Parameter Name	Type	M/ O/C	Description
connectionToSendDigits	ConnectionID	M	Connection of the device which is generating the digits for the call.
digitMode	Enumerated	O	Specifies the signalling format. The complete set of possible values is: • rotaryPulse - rotary pulse signalling. • DTMF - DTMF signalling (default).
charactersToSend	Characters (64)	M	Specifies the string of characters to send. Shall consist of the following set: • For rotary pulse digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D • For DTMF digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, *, #, A, B, C, D A comma “,” may be included in the parameter string to indicate a pause between characters. The length of the pause is switching function specific and may be determined using the capabilities exchange services. This parameter type is a character string. The maximum length supported by the switching function is provided via the capabilities exchange services.
toneDuration	Value	O	Specifies the duration, in milliseconds, of each tone to be provided (valid for DTMF digitMode only).
pulseRate	Value	O	Specifies the number of pulses per second (valid for rotaryPulse digitMode only).
pauseDuration	Value	O	Specifies the duration, in milliseconds, of the silences between the digits (interdigit delay).

Table 18-10 Generate Digits—Service Request (continued)

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

18.1.3.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

18.1.3.2.1 Positive Acknowledgement

Table 18-11 Generate Digits—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

Note that the positive acknowledgement does not necessarily mean that the charactersToSend have been successfully received by a device in the call.

18.1.3.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

18.1.3.3 Operational Model

18.1.3.3.1 Connection State Transitions

There are no connection state changes due to this service.

18.1.3.3.2 Device-Type Monitoring Event Sequences

Table 18-12 Generate Digits—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event
D1	D1C1 (connectionToSendDigits)	Digits Generated

18.1.3.3.3 Call-Type Monitoring Event Sequences

Table 18-13 Generate Digits—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event
C1 (existingCall)	D1C1 (connectionToSendDigits)	Digits Generated

18.1.3.3.4 Functional Requirements

1. This service is used for end-to-end signalling purposes. This service shall not be used to perform dialling sequences (the Dial Digits service shall be used to perform this function).
2. The charactersToSend parameter may contain additional non-standardised characters. Handling of the additional characters is switching function dependent.
3. If the toneDuration or pulseRate parameters in the service request are not present or if they are present but cannot be satisfied, the switching function may use its default value.
4. A Generate Digits service request may be rejected if there is already a switching function generated tone (busy, music, ringback, etc.) on the call.
5. If a Generate Digits service request is received by the switching function while a previous Generate Digits service request has not yet been completed, the new request may or may not be rejected but in any case the digits shall not be interleaved or overlaid.

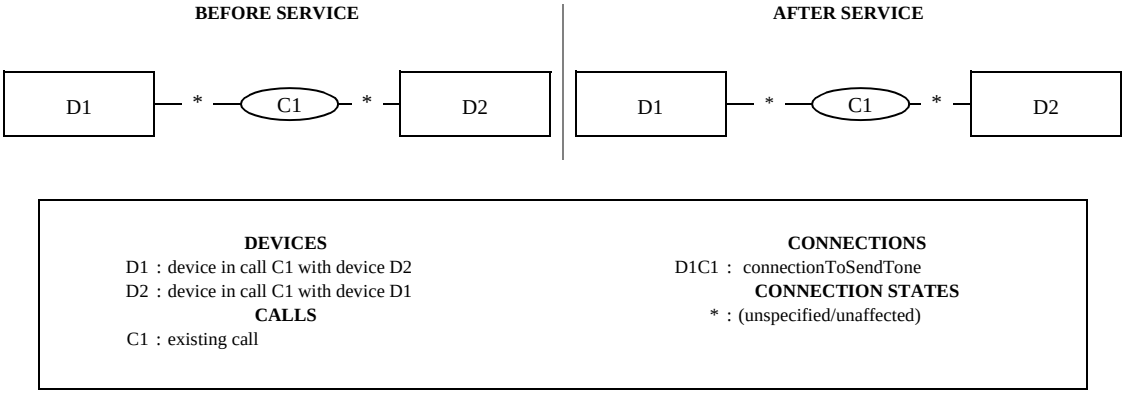
18.1.4 Generate Telephony Tones

C → S

The Generate Telephony Tones service causes a telephony tone such as a beep, busy, or ringback to be sent on behalf of a connection in a call. This service also supports optional parameters to control tone generation.

This service does not affect the state or progress of a call.

Figure 18-4 Generate Telephony Tones Service



18.1.4.1 Service Request

Table 18-14 Generate Telephony Tones—Service Request

Parameter Name	Type	M/ O/C	Description
connectionToSendTone	ConnectionID	M	Connection of the device which is generating the telephony tones for the call
toneToSend	Enumerated	M	Specifies the tone to send. The complete set of possible values is: • beep • billing • busy • carrier • confirmation • dial • faxCNG • hold • howler • intrusion • modemCNG • park • record warning (indicates call may be being recorded) • reorder • ringback • silence • SIT VC • SIT IC • SIT RO • SIT NC • SwitchSpecified0 through SwitchSpecified100 - reserved for switch specified tones.

Table 18-14 Generate Telephony Tones—Service Request (continued)

Parameter Name	Type	M/ O/C	Description
toneDuration	Value	O	Specifies the total length in milliseconds of the tones to be provided. A value of 0 indicates that the tone should be continuous.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

18.1.4.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

18.1.4.2.1 Positive Acknowledgement

Table 18-15 Generate Telephony Tones—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

Note that the positive acknowledgement indicates that the requested tone is successfully being presented.

18.1.4.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

18.1.4.3 Operational Model

18.1.4.3.1 Connection State Transitions

There are no connection state changes due to this service.

18.1.4.3.2 Device-Type Monitoring Event Sequences

Table 18-16 Generate Telephony Tones—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event
D1	D1C1 (connectionToSendTone)	Telephony Tones Generated

18.1.4.3.3 Call-Type Monitoring Event Sequences

Table 18-17 Generate Telephony Tones—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event
C1 (existingCall)	D1C1 (connectionToSendTone)	Telephony Tones Generated

18.1.4.3.4 Functional Requirements

1. This service is used to send inband tones for end-to-end signalling purposes.
2. The toneToSend parameter specifies standard tones. The switching function provides the exact tone based upon many factors such as country and configuration information.
3. A Generate Telephony Tones service request may be rejected if there is already a switching function generated tone (busy, music, ringback, etc.) on the call.
4. If the toneDuration parameter is not present in the service request, or if it is present but cannot be satisfied, the switching function may use its default value.

5. If a Generate Telephony Tones service request is received by the switching function while a previous Generate Digits service request has not yet been completed, the new request may or may not be rejected, but in any case the digits shall not be interleaved or overlaid.
6. The requested tone continues to be generated until the specified duration expires, until it is canceled by using the Cancel Telephony Tones service, or until the call is cleared.

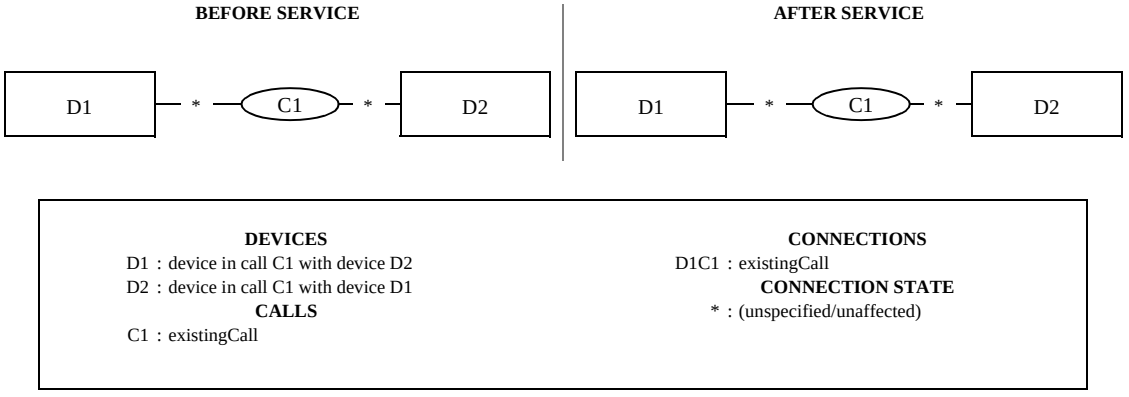
18.1.5 Send User Information

C → S

The Send User Information service is used to send user data information.

This service does not affect the state or progress of a call.

Figure 18-5 Send User Information Service



Note that the existingCall connectionID could either be the D1C1 or the D2C1 connections in the above figure.

18.1.5.1 Service Request

Table 18-18 Send User Information—Service Request

Parameter Name	Type	M/ O/C	Description
existingCall	ConnectionID	M	Specifies the connection on whose behalf user data is to be sent.
userData	UserData	M	Specifies the user data to send.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

18.1.5.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

18.1.5.2.1 Positive Acknowledgement

Table 18-19 Send User Information—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

Note that the positive acknowledgement does not necessarily mean that the user information has been successfully received by the other device in the call. It only indicates that user information has been sent.

18.1.5.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

18.1.5.3 Operational Model

18.1.5.3.1 Connection State Transitions

There are no connection state changes due to this service.

18.1.5.3.2 Device-Type Monitoring Event Sequences

Table 18-20 Send User Information—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event
D1	D1C1	Call Information
D2	D1C1	Call Information

18.1.5.3.3 Call-Type Monitoring Event Sequences

Table 18-21 Send User Information—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event
C1	D1C1	Call Information

18.1.5.3.4 Functional Requirements

1. The Send User Information is used to send information end to end. The user data provided in the Send User Information service is not managed or maintained by the switching function, but instead transmitted to all devices in the call. For a complete description of the behaviour of user data, refer to 6.1.4.4, “User Data”, on page 29.
2. The ability to send user information, the timing of when the information can be sent, and the size of the data, depends upon the switching function’s capabilities and the underlying network (such as ISDN).
3. The Call Information event is used to provide the user data information received on a call.

18.2 Events

Table 18-22 Call Associated Feature Events Summary

Call Associated Feature Event	Description	Pg.
18.2.1 Call Information	Indicates that call associated information (such as correlator data, account code, authorisation code, call qualifying data, etc.) has been collected for a call.	344
18.2.2 Charging	Indicates that new charging information has arrived for a device involved in a call.	346
18.2.3 Digits Generated	Indicates that (DTMF or rotary pulse) digits have been generated.	347
18.2.4 Telephony Tones Generated	Indicates that telephony tones have been generated.	348
18.2.5 Service Completion Failure	Indicates that a previous multi-step computing function initiated service request has failed before that service's successful completion conditions were satisfied.	351

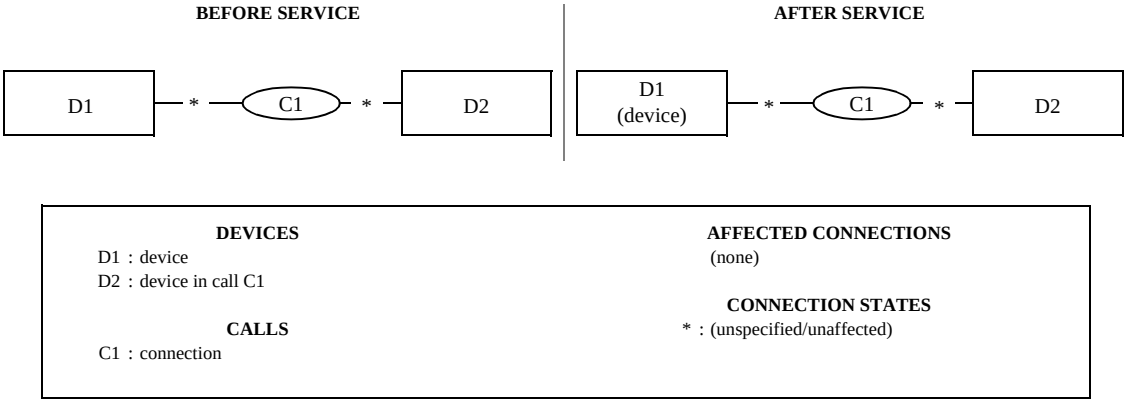
18.2.1 Call Information

The Call Information event indicates that call associated information has been collected/updated for a call.

Common situations that generate this event include:

- The account code or authorization code has been included or changed via a call control service request, or has been manually entered.
- The Associate Data service has been used to associate information with a call.
- User data or correlator data has been received independent of call activity (via the Send User Information service or as a result of non-call related activity from an external network).
- CallerID information associated with the call has been updated.
- Services permitted information has been updated as a result of another connection changing state.

Figure 18-6 Call Information Event



Note that in some situations, there may only be one device (D1) in the call.

18.2.1.1 Event Parameters

Table 18-23 Call Information—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
connection	ConnectionID	M	Indicates the connection of the device responsible for associating the information with the call.
device	SubjectDeviceID	M	Indicates the device responsible for associating the information with the call.
callingDevice [1]	CallingDeviceID	C	Indicates the original calling device associated with the call
accountInfo [1]	AccountInfo	C	Indicates the account code associated with the call.
authorisationCode [1]	AuthCode	C	Indicates the authorization code associated with the call.
correlatorData [1]	CorrelatorData	C	Indicates the correlator data associated with the call.
servicesPermitted [1]	ServicesPermitted	C	Provides a list of services that can be applied to the specified connection.
userData [1]	UserData	C	Provides information that has been sent from another device.
callQualifyingData [1]	CallQualifyingData	C	Indicates the call qualifying data (wrap code, walk away code, etc.) associated with the call.
connectionInfo [1]	ConnectionInformation	C	Indicates the connection information associated with the connection.

Table 18-23 Call Information—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Non-standardized information with the event.

1. At least one of these service specific parameters shall be provided on this event.

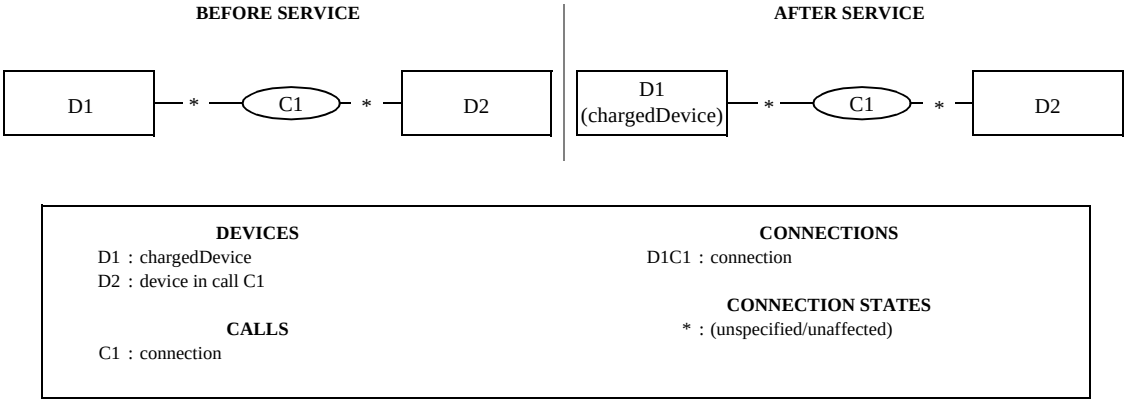
18.2.1.2 Functional Requirements

1. When either the accountInfo or callQualifyingData parameters is provided, the callID component of the connectionID in the event may be omitted or may refer to the connectionID that no longer exists at the device. This may occur when a user enters a wrap code for a call that has just cleared, for example.
2. This event is generated when the switching function receives user data independent of call activity as a result of the Send User Information service or as a result of non-call activity from an external network, for example.
3. This event is generated when correlator data is created or changed independently of a call control service. It is not generated when correlator data is passed with a Call Control service request (in this case, the new correlator data is presented to the computing function through the appropriate call control event).
4. This event is generated when connection information (number of media stream channels, for example) is changed independently of a call control service. It is not generated when connection information is passed with a Call Control service request (in this case, the new connection information is presented to the computing function through the appropriate call control event).
5. If the Associate Data service is used to associate data with a call, the Call Information event is generated.
6. If the list of CSTA services that can be applied to a connection changes as the result of another call’s connection changing state, for example, the Call Information event is generated with the updated servicesPermitted parameter associated with the specified connection that did not change state.

18.2.2 Charging

The Charging event indicates that new call charging information (such as a network provided Periodic Pulse Metering (PPM) signal) has been detected for a device involved in a call. This event may occur after the call has been released, to report arrival of additional call charge information.

Figure 18-7 Charging Event



18.2.2.1 Event Parameters

Table 18-24 Charging—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
connection	ConnectionID	M	Indicates the call for which the device is being charged.
chargedDevice	DeviceID	M	Indicates the device that is being charged. If this device is not specified then this parameter indicates “Unknown” or “Not Required”.
chargingInfo	ChargingInfo	M	Indicates an intermediate or total value of charging or currency units for the device being charged.
cause	EventCause	O	Indicates a reason for the event.
security	CSTASecurityData	O	Indicates timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Indicates the non-standardized information

18.2.2.2 Event Causes

Table 18-25 Charging—Event Causes

Event Cause	Description	Associated Features
Network Signal	The event was generated as the result of a particular network event or condition.	External Call
Normal	Charging information has been detected.	External Call
Timeout	The event was generated because of a timeout.	External Call

18.2.2.3 Functional Requirements

1. The frequency with which this event is generated depends upon the implementation of the call charge information providing service outside the switching sub-domain. It also may depend upon the switching function implementation.

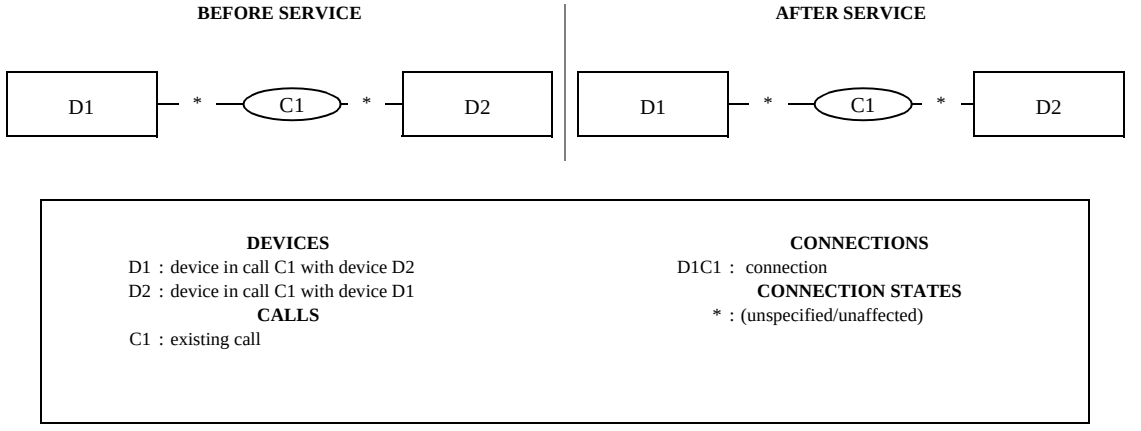
18.2.3 Digits Generated

The Digits Generated event indicates that DTMF or rotary pulse digits have been generated at a device.

Common situations that generate this event include:

- The switching function generates DTMF or rotary pulse digits for the device of a given connection as the result of the Generate Digits service or through manual activity.

Figure 18-8 Digits Generated Event



18.2.3.1 Event Parameters

Table 18-26 Digits Generated—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
connection	ConnectionID	M	The connection at the device
digitGeneratedList	Characters (64)	M	The sequence of digits generated. • For rotary pulse digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D • For DTMF digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, *, #, A, B, C, D This parameter shall only include previously unreported digits.
digitDurationList	List of Values	O	The list of corresponding digit durations, in milliseconds for each generated digit.
pauseDurationList	List of Values	O	The list of corresponding pauses, in milliseconds, before each generated digit. The duration of the pause before the first digit is zero.
connectionInfo	ConnectionInformation	O	Specifies the connection information associated with the connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

18.2.3.2 Functional Requirements

1. A Digits Generated event is sent when the device of the specified connection generates digits (via the Generate Digits service or through manual activity).
2. The Data Collection services (Clause 25, “Data Collection Services”) are used to report DTMF or rotary pulse digits that are received over a connection at a device.

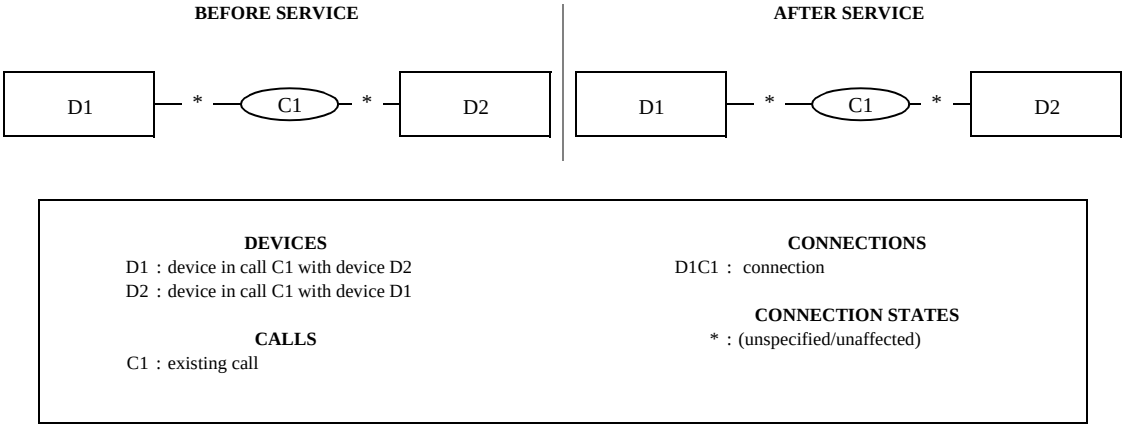
18.2.4 Telephony Tones Generated

The Telephony Tones Generated event indicates that telephony tones have been generated at a device.

Common situations that generate this event include:

- The switching function generates telephony tones (via the Generate Telephony Tones service, for example) for the device of a given connection.

Figure 18-9 Telephony Tones Generated Event



18.2.4.1 Event Parameters

Table 18-27 Telephony Tones Generated—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
connection	ConnectionID	M	The connection at the device

Table 18-27 Telephony Tones Generated—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
toneGenerated	Enumerated	C	Specifies the generated tone. The complete set of possible values is: • beep • billing • busy • carrier • confirmation • dial • faxCNG • hold • howler • intrusion • modemCNG • park • record warning (indicates call may be being recorded) • reorder • ringback • silence • SIT VC • SIT IC • SIT RO • SIT NC • SwitchSpecified0 through SwitchSpecified100 - reserved for switch specified tones. • other • unknown This parameter shall not be provided when a previously reported tone has stopped being generated.
toneFrequency	Value	O	Specifies (in Hz) the switching function determined frequency of the generated tone. This parameter shall not be provided if the toneGenerated is anything other than “other”.
toneDuration	Value	O	The duration, in milliseconds, of the tone. This parameter shall not be provided if the toneGenerated is anything other than “other”.
pauseDuration	Value	O	The pause, in milliseconds, between the last tone and this one. This parameter shall not be provided if the toneGenerated is anything other than “other”.
connectionInfo	ConnectionInformation	O	Specifies the connection information associated with the connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information

18.2.4.2 Functional Requirements

1. The Telephony Tones Generated event is sent when the device of the specified connection generates telephony tones (via the Generate Telephony Tones service).
2. When the generated telephony tone is not longer generated (due to the Cancel Telephony Tones service, for example) another Telephony Tones Generated event may be sent (if the switching function supports the generation of the event when a telephony tone is stopped) to indicate that the telephony tone is no longer being generated (with the toneGenerated parameter not provided).
3. The toneFrequency, toneDuration, and pauseDuration parameters can only be provided when the generatedTone is “other”.
4. The services described in Clause 25, “Data Collection Services” are used to report telephony tones that are received over a connection at a device.

18.2.5 Service Completion Failure

The Service Completion Failure event indicates that a previous multi-step computing function initiated service request (as indicated by the switching function in the capabilities exchange services) has failed before that service's successful completion conditions were satisfied.

It is the responsibility of the computing function to monitor for this event, and any other event associated with a particular multi-step service request, in order to determine whether or not the request has successfully met its completion conditions.

18.2.5.1 Event Parameters

Table 18-28 Service Completion Failure—Event Parameters

Parameter Name	Type	M/ O/C	Description
monitorCrossRefID	MonitorCrossRefID	M	Associates the event to an established monitor.
primaryCall	Structure	M	Specifies information for the connection in the primary call that is associated with the service request. This includes: • deviceID (M) DeviceID - a device involved in the call. • connectionID (M) ConnectionID - a connection at the device. • localConnectionState (M) LocalConnectionState - the local connection state of the connection. • connectionInformation (O) ConnectionInformation - the connection information for the connection.
secondaryCall	Structure	O	Specifies information for the connection in the secondary call that is associated with the service request. This includes: • deviceID (M) DeviceID - a device involved in the call. • connectionID (M) ConnectionID - a connection at the device. • localConnectionState (M) LocalConnectionState - the local connection state of the connection. • connectionInformation (O) ConnectionInformation - the connection information for the connection. This parameter may only be present if the service request involved a secondary call.
otherDevicesPrimaryCallList	List of Structures	O	Specifies information for each connection in the primary call. This includes: • deviceID (M) DeviceID - a device involved in the call. • connectionID (M) ConnectionID - a connection at the device. • localConnectionState (O) LocalConnectionState - the local connection state of the connection. • connectionInformation (O) ConnectionInformation - the connection information for the connection. This parameter may only be present if the other connections in the primary call still exist. This parameter may not contain the information associated with the primaryCall parameter.

Table 18-28 Service Completion Failure—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
otherDevicesSecondaryCall List	List of Structures	O	Specifies information for each of the other connection in the secondary call. This includes: • deviceID (M) DeviceID - a device involved in the call. • connectionID (M) ConnectionID - a connection at the device. • localConnectionState (O) LocalConnectionState - the local connection state of the connection. • connectionInformation (O) ConnectionInformation - connection information for the connectionID. This parameter shall only be present if the service request involved a secondary call and the other connections associated with the secondary call still exist. This parameter shall not contain the information associated with the secondaryCall parameter.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call.
cause	EventCause	M	Indicates the reason why the service request did not complete.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information with the event.

18.2.5.2 Event Causes

Table 18-29 Service Completion Failure—Event Causes

Event Cause	Description	Associated Features
Blocked	The call failed after a device has disconnected from a call leaving one other device remaining in the call.	Service Request Failure
Busy	The call encountered a busy or unavailable device.	Service Request Failure
Call Cancelled	The call has been terminated before the associated device has gone on-hook.	Service Request Failure
Call Not Answered	The call was not answered because a timer elapsed.	Service Request Failure
Destination Not Obtainable	The call could not reach the destination.	Service Request Failure
Destination Out of Order	The call failed because it encountered a destination out of service.	Service Request Failure
Do Not Disturb	The call failed because it encountered a device that has the do not disturb feature set.	Service Request Failure
Incompatible Destination	The call encountered an incompatible destination.	Service Request Failure
Invalid Account Code	The call has an invalid account code.	Service Request Failure
Invalid Number Format	The call failed because the dialled number is incorrect	Service Request Failure
Key Operation in Use	The call failed because an appearance is associated with an exclusive bridged device configuration and that the appearance is disabled until the call is terminated or until the call moves away from the device.	Service Request Failure
Lockout	The call encountered inter-digit time-out while dialling.	Service Request Failure
Maintenance	The call encountered a facility or device in a maintenance condition.	Service Request Failure
Network Congestion	The call encountered a congested network.	Service Request Failure
Network Not Obtainable	The call could not reach a destination network.	Service Request Failure
Network Signal	The call failed because it encountered a problem after it left the switching sub-domain.	Service Request Failure

Table 18-29 Service Completion Failure—Event Causes (continued)

Event Cause	Description	Associated Features
No Available Agents	The call could not access any agent.	Service Request Failure
Not Available Bearer Service	The call failed because it was requested with a bearer capability that is currently not available.	Service Request Failure
Not Supported Bearer Service	The call failed because it was requested with a bearer capability that is currently not supported.	Service Request Failure
Number Changed	The called number has been changed to a new number.	Service Request Failure
Number Unallocated	The call failed because the called number is not allocated to a subscriber.	Service Request Failure
Reorder Tone	The call encountered a reorder condition.	Service Request Failure
Resources not Available	Resources were not available.	Service Request Failure
Selected Trunk Busy	The call failed because a specific selected Network Interface Device (e.g., trunk, CO line) is busy.	Service Request Failure
Trunks Busy	The call encountered Trunks Busy.	Service Request Failure
Unauthorized Bearer Service	The call failed because it was requested with an unauthorized bearer capability.	Service Request Failure

18.2.5.3 Functional Requirements

1. This event is only reported to device-type monitors on the device which has or had connection(s) that were used in the particular service request which is associated with this event (i.e., this event is not reported to any call-type monitors or device-type monitors for other devices in the call(s) associated with the service request).
2. If the switching function supports this event, it shall be provided on all multi-step service requests (supported by the switching function) if the completion conditions is not successfully met.
3. The Service Completion Failure event shall not be provided by the switching function to indicate a connection’s transition to the Fail state. A Failed event shall be provided to indicate this.
4. If event flow, prior to receiving the Service Completion Failure event, has indicated that the service request has caused a state change, it is up to the computing function to apply the appropriate service(s) to return the call(s) and/or device(s) back to their original conditions (if needed and possible).

19 Media Attachment Services & Events

This clause describes services and events which enable access to the media stream of a call through attachment and detachment of media services to the call.

19.1 Services

Table 19-1 Media Attachment Services Summary

Media Attachment Service	Description	Pg.
19.1.1 Attach Media Service	Attaches a media service instance to a call.	355
19.1.2 Detach Media Service	Detaches a media service instance from a call.	359

19.1.1 Attach Media Service

C → S

The Attach Media Service service attaches an existing call to a media service instance for the purpose of transmitting media stream data to the call or receiving media stream data from the call.

The attachment is made at a connection associated with a media attachment device (MAD). The attachment may be performed by the switching function selecting a MAD and adding it to the call by means of a call control service(s) (e.g., Conference Call, Transfer Call, Deflect Call, Directed Pickup Call, Join Call), or may be performed via a device already in the call (that has media access capabilities). At the completion of this service request a Media Attached event is generated on monitors associated with the specified call or device. This event may contain the associated mediaStreamID for accessing the media service instance.

The service request specifies a media service instance by choosing appropriate values for the mediaServiceType and mediaServiceInstanceID parameters, and a connectionMode parameter that specifies how the attachment is to be made.

The roles played by the various devices in the call, and the means by which the attachment is performed, depend upon the connectionMode. The connectionMode consists of a set of values that specify the call control service(s) that is to be applied to the connection so that the MAD can be added to the call, or the value of direct, which specifies that the MAD is already part of the call.

Refer to the specification of the call control service(s) associated with the specified connectionMode for a description of the operational model.

19.1.1.1 Service Request

Table 19-2 Attach Media Service—Service Request

Parameter Name	Type	M/ O/C	Description
connection	ConnectionID	M	Specifies the connection at the attaching device.
mediaServiceType	MediaServiceType	M	Specifies the requested media service type.
mediaServiceVersion	Value	O	Specifies the version of the media services.
mediaServiceInstanceID	MediaServiceInstanceID	O	Specifies the desired media service instance.

Table 19-2 Attach Media Service—Service Request (continued)

Parameter Name	Type	M/ O/C	Description
connectionMode	Enumerated	M	Specifies the requested media service connection mode. The complete set of possible values is: • consultationConference - conference the media access device (MAD) into the call via the Consultation Call and Conference Call models. • consultationConferenceHold - conference the MAD into the call via a Consultation Call and Hold Call models (the connection at the attaching device is placed on hold). • deflect - move the call from the attaching device to the MAD via the Deflect Call model. • directedPickup - direct the call from the attaching device to the MAD via the Directed Pickup Call model. • join - add a device into the call via the Join Call model. The application chooses which device is the MAD in the resulting conference. • singleStepConference - conference the MAD into the call via the Single Step Conference Call models. • singleStepConferenceHold - conference the MAD into the call via a Single Step Conference Call and Hold Call models (the connection at the attaching device is placed on hold). • singleStepTransfer - transfer that MAD into the call via the Single Step Transfer Call model. • transfer - transfer the MAD into the call via the Consultation Call and Transfer Call models. • direct - bind the media service instance directly to the existing connection at the attaching device.
requestedConnectionState	LocalConnectionState	O	Specifies the connection state in which the MAD shall be at the time that the Media Attached event is generated.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

19.1.1.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

19.1.1.2.1 Positive Acknowledgement

Table 19-3 Attach Media Service—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
mediaConnection	ConnectionID	C	Specifies the connection at the media access device. See Functional Requirement #7.
mediaDevice	DeviceID	C	Specifies the device identifier of the media access device. See Functional Requirement #7.
mediaServiceInstanceID	MediaServiceInstanceID	O	Specifies the media service instance used.
mediaConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the mediaConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

19.1.1.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

19.1.1.3 Operational Model

19.1.1.3.1 Connection State Transitions

Refer to the specification of the call control service(s) associated with the specified connectionMode for a description of the connection state transitions.

19.1.1.3.2 Device-Type Monitoring Event Sequences

Table 19-4 Attach Media Service—Device-Type Monitoring Event Sequences

Monitored Device	Connection	Event	Event Cause
(media access device)	DxCx	Call Control events depend upon the connectionMode parameter.	
	DxCx	Media Attached	(not applicable)
(other devices in the call)	DxCx	Call Control events depend upon the connectionMode parameter.	
	DxCx	Media Attached	(not applicable)

19.1.1.3.3 Call-Type Monitoring Event Sequences

Table 19-5 Attach Media Service—Call-Type Monitoring Event Sequences

Monitored Call	Connection	Event	Event Cause
Cx	DxCx	Call Control events depend upon the connectionMode parameter.	
	DxCx	Media Attached	(not applicable)

Note that the Media Attached event may be generated before other call control events associated with adding the media access device to the call.

19.1.1.3.4 Functional Requirements

1. If the computing function is monitoring the call or any devices in the call, it shall be prepared to receive the call control events, if any, associated with adding the media access device to the call (e.g., Conferenced, Transferred, Held) events.
2. The specification of the call control service(s) (service description, operational model, functional requirements, etc.) that are described by the connectionMode parameter apply to the Attach Media service.

3. Multiple attachments to media services for the same call and connection are possible. However, implementations may place limits on the number of such attachments. Each additional attachment may or may not involve a conference/transfer operation to take place (as in the case of the first attachment). If the computing function tries to attach a mediaServiceType to a connection that already has that mediaServiceType attached, the request will be rejected by the switching function.
4. Authentication with the media service is not implied by a positive acknowledgement to an Attach Media Service service request. The computing function shall open a connection to and authenticate with the media service through the appropriate mechanisms.
5. Versioning of the media service is supported via a separate mediaServiceVersion parameter. With respect to these services, however, two different versions of the same media service are treated as if they were altogether two different media services.
6. The switching function may place restrictions on the services that may be applied to a media access device. As a result, the computing function should always use the Detach Media Service service to release the binding of the media service instance from the call rather than trying to manipulate the media access device directly using other call control services.
7. The mediaConnection and mediaDevice parameters in the positive acknowledgement to this service shall be provided only if the connectionMode parameter in the service request is any value other than direct.

19.1.2 Detach Media Service

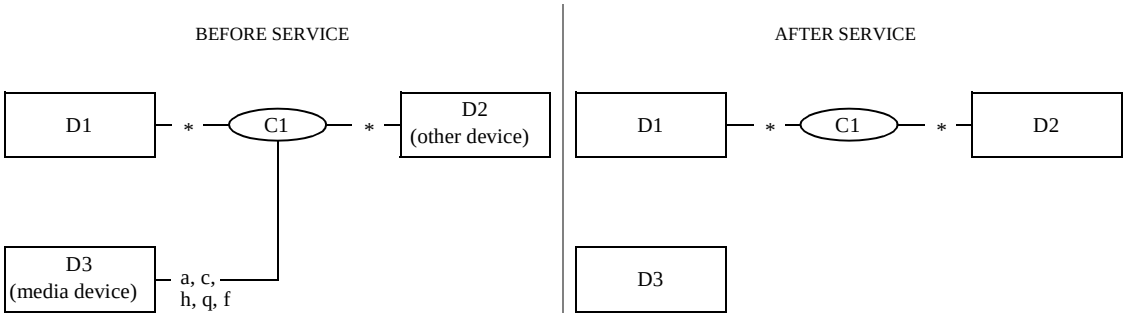
C → S

The Detach Media Service service removes a previously attached media service instance from a call. This request releases a previously established association (binding) between a connection and a media service. If a media access device was added to the call when the media service instance was attached and that media access device is no longer needed, then the media access device is removed (i.e., cleared) from the call.

There are two cases for the Detach Media Service service request:

- Case A: A media access device (D3) was previously connected into the call using the Attach Media Service service. In this case the media access device is removed from the call by this service. Figure 19-1 is an example where a media access device that was conferenced into a call is detached from a call.

Figure 19-1 Detach Media Service (Case A)



DEVICES
D1 : attaching device
D2 : other device in call C1
D3 : media access device

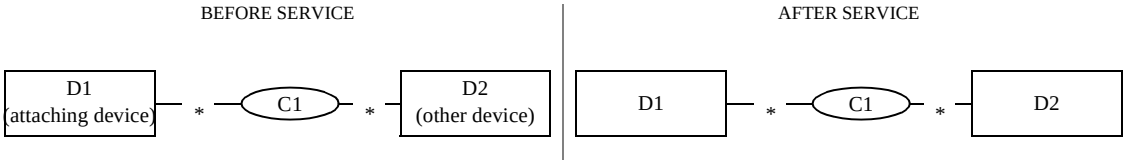
CALLS
C1 : existing call (connection)

AFFECTED CONNECTIONS
D3C1 : media connection

CONNECTION STATES
a : alerting state
c : connected state
h : hold state
q : queued state
f : fail state
* : (unspecified/unaffected)

- Case B: The media service instance was directly bound to an existing connection (D1C1) in the call. In this case this detaching device remains in the call after the service request.

Figure 19-2 Detach Media Service (Case B)



DEVICES
D1 : detaching device
D2 : other device in call C1

CALLS
C1 : connection (existing call)

AFFECTED CONNECTIONS
D1C1 : detaching connection (existing media connection)

CONNECTION STATES
* : (unspecified/unaffected)

19.1.2.1 Service Request

Table 19-6 Detach Media Service—Service Request

Parameter Name	Type	M/ O/C	Description
connection	ConnectionID	M	Specifies the connection at the media access device, if used. Otherwise, specifies the connection at the detaching device.
mediaServiceType	MediaServiceType	M	Specifies the requested media service type to detach.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

19.1.2.2 Service Response

The capability exchange services describe the type of acknowledgement model (atomic or multi-step) that the switching function supports for this service request.

19.1.2.2.1 Positive Acknowledgement

Table 19-7 Detach Media Service—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

19.1.2.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

19.1.2.3 Operational Model

19.1.2.3.1 Connection State Transitions

The following tables describes the connections that are affected by this service.

- Case A: A previously added media access device (D3) is removed from the call.

Table 19-8 Detach Media Service—Connection State Transitions (Case A)

Connection	Initial State (Required)	Final State
D1C1	(Ignored)	(Unaffected; no transition due to this service)
D2C1	(Ignored)	(Unaffected; no transition due to this service)
D3C1 (media connection)	Alerting, Connected, Hold, Queued, Fail	Null

- Case B: The media service instance is unbound from a detaching device (D1) in the call.

Table 19-9 Detach Media Service—Connection State Transitions (Case B)

Connection	Initial State (Required)	Final State
D1C1	(Ignored)	(Unaffected; no transition due to this service)
D2C1	(Ignored)	(Unaffected; no transition due to this service)

19.1.2.3.2 Device-Type Monitoring Event Sequences

The following table describes the connections that are affected by this service.

- Case A: A previously added media access device (D3) is removed from the call.

Table 19-10 Detach Media Service—Device-Type Monitoring Event Sequences (Case A)

Monitored Device	Connection	Event	Event Cause
D1	D3C1	Media Detached	(not applicable)
	D3C1	Connection Cleared	Normal Clearing
D2	D3C1	Media Detached	(not applicable)
	D3C1	Connection Cleared	Normal Clearing
D3 (media access device)	D3C1	Media Detached	(not applicable)
	D3C1	Connection Cleared	Normal Clearing

- Case B: The media service instance is unbound from a detaching device (D1) in the call.

Table 19-11 Detach Media Service—Device-Type Monitoring Event Sequences (Case B)

Monitored Device	Connection	Event	Event Cause
D1 (detaching device)	D1C1	Media Detached	(not applicable)
D2 (other device)	D1C1	Media Detached	(not applicable)

19.1.2.3.3 Call-Type Monitoring Event Sequences

- Case A: A previously added media access device (D3) is removed from the call.

Table 19-12 Detach Media Service—Call-Type Monitoring Event Sequences (Case A)

Monitored Call	Connection	Event	Event Cause
C1	D3C1	Media Detached	(not applicable)
	D3C1	Connection Cleared	Normal Clearing

- Case B: The media service instance is unbound from a detaching device (D1) in the call.

Table 19-13 Detach Media Service—Call-Type Monitoring Event Sequences (Case B)

Monitored Call	Connection	Event	Event Cause
C1 (existing call)	D1C1	Media Detached	(not applicable)

19.2 Events

Table 19-14 Media Attachment Events Summary

Media Attachment Event	Description	Pg.
19.2.1 Media Attached	Indicates that a media service instance has been attached to a call.	363
19.2.2 Media Detached	Indicates that a media service instance has been detached from a call.	364

19.2.1 Media Attached

The Media Attached event indicates an attachment of media services to a call. This event may be generated as a result of a previous Attach Media Service service request, or as a result of an automatic media service attachment.

Refer to the specification of the call control services(s) and resulting call control events(s) associated with the specified connectionMode for a description of the operational model.

19.2.1.1 Event Parameters

Table 19-15 Media Attached—Event Parameters

Parameter Name	Type	M/ O/C	Description
mediaConnection	ConnectionID	M	Specifies the connection that was bound to the media service
mediaDevice	SubjectDeviceID	M	Specifies the device identifier associated with the attached media service.
mediaServiceType	MediaServiceType	M	Specifies the media service type.
mediaServiceVersion	Value	O	Specifies the version of the media services.
mediaServiceInstanceID	MediaServiceInstanceID	O	Specifies the media service instance associated with the attached media service.
mediaStreamID	MediaStreamID	C	Specifies the Media stream ID used to access the media service. This parameter shall be provided if the switching function supports providing the mediaStreamID as indicated by the capability exchange services.
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. This parameter is mandatory for events generated for device-type monitors and otherwise shall not be provided.
mediaConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the mediaConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

19.2.1.2 Functional Requirements

1. If a media access device was added to the call as a result of a successful Attach Media Service service request, then the mediaConnection and mediaDevice parameters refer to the media access device. If the media service instance was directly bound to an existing connection in the call, then the mediaConnection and mediaDevice parameters refer to the connection at that attaching device.

19.2.2 Media Detached

The Media Detached event indicates a detachment of media services from a connection. This event may be generated as a result of a previous Detach Media Service service request, or as a result of an automatic media service detachment.

There are two cases for the Media Detached event:

- Case A: A media access device (D3) was previously connected into the call using the Attach Media Service service. In this case the media access device is removed from the call by this service.
- Case B: The media service instance was directly bound to an existing connection (D1C1) in the call. In this case this detaching device remains in the call after the service request.

Figure 19-3 Media Detached Event (Case A)

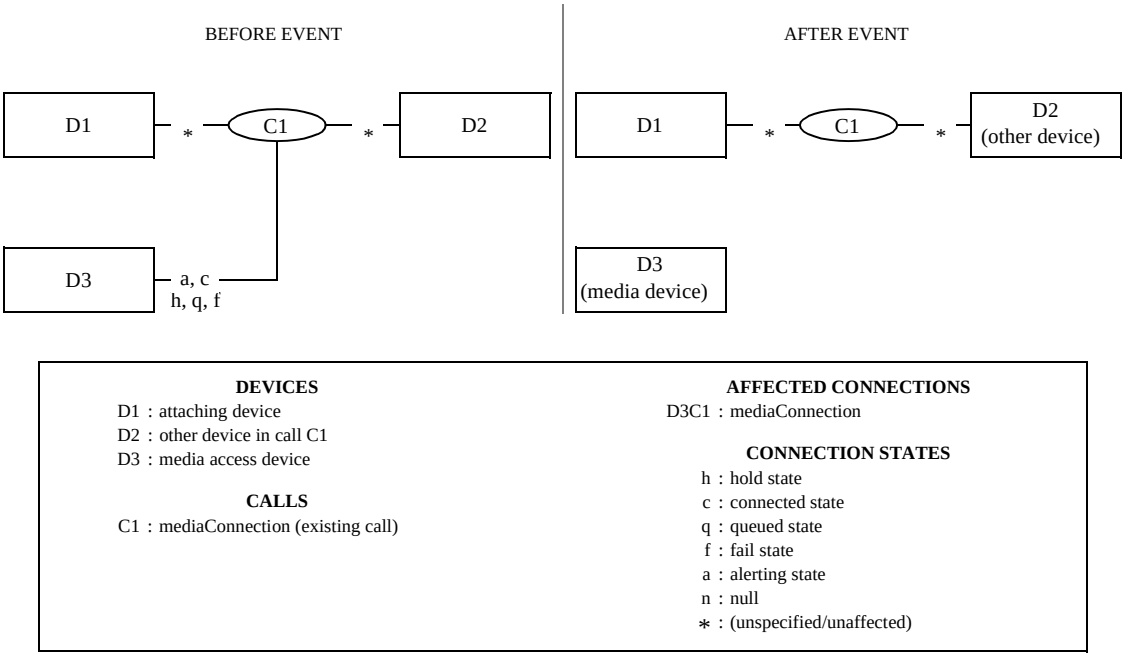
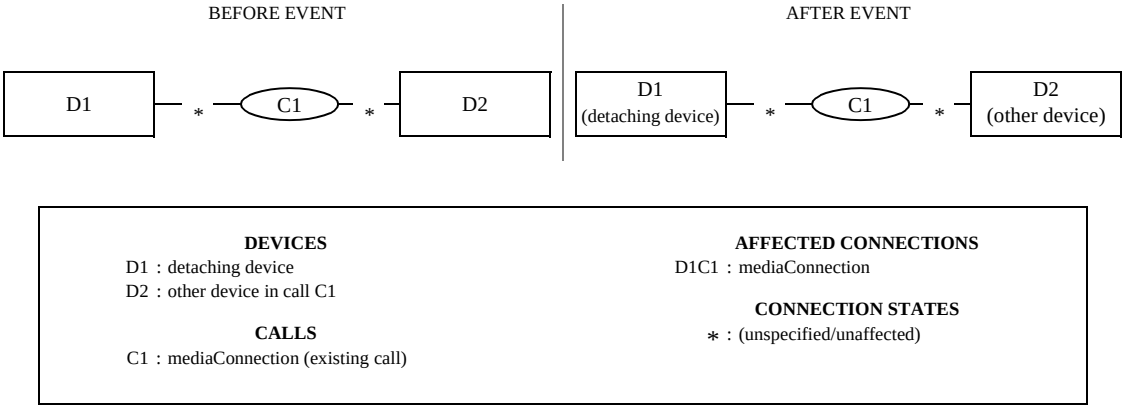


Figure 19-4 Media Detached Event (Case B)



19.2.2.1 Event Parameters

Table 19-16 Media Detached—Event Parameters

Parameter Name	Type	M/ O/C	Description
mediaConnection	ConnectionID	M	Specifies the connection that was unbound from the media service.

Table 19-16 Media Detached—Event Parameters (continued)

Parameter Name	Type	M/ O/C	Description
mediaDevice	SubjectDeviceID	M	Specifies the device identifier associated with the detached media service.
mediaServiceType	MediaServiceType	M	Specifies the media service type detached.
mediaServiceVersion	Value	O	Specifies the version of the media services.
mediaServiceInstanceID	MediaServiceInstanceID	O	Specifies the media service instance associated with the detached media service. See Functional Requirement #
mediaStreamID	MediaStreamID	C	Specifies the Media stream ID associated with the media service that was detached. This parameter shall be provided if the switching function supports providing the mediaStreamID as indicated by the capability exchange services
mediaCallCharacteristics	MediaCallCharacteristics	O	Specifies the media class and media characteristics of the call. See 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete description.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4, “CallCharacteristics”, on page 84 for the complete set of possible values.
localConnectionInfo	LocalConnectionState	C	Specifies the local connection state of the device associated with the Monitor Cross Reference ID. This parameter is mandatory for events generated for device-type monitors and otherwise shall not be provided.
mediaConnectionInfo	ConnectionInformation	O	Specifies the connection information associated with the mediaConnection connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

19.2.2.2 Functional Requirements

1. If a media access device was added to the call as a result of a successful Attach Media Service, then the mediaConnection and mediaDevice parameters refer to the media access device. If the media service instance was directly bound to an existing connection in the call, then the mediaConnection and mediaDevice parameters refer to the connection at the attaching device.
2. The switching function may initiate a media service detachment whenever it is unable to maintain the binding of the connection to the media service. This may occur as a result of other activity on the call.

20 Routeing Services

This section specifies two types of Routeing services:

- Route Registration services
- Call Routeing services

NOTE

This clause describes Routeing services between the Switching Function and the Computing Function.

20.1 Registration Services

Table 20-1 Route Registration Services Summary

Route Registration Service	Description	Pg.
20.1.1 Route Register	Registers the computing function as a routeing server for a specified routeing device or for the entire switching function.	367
20.1.2 Route Register Abort	Specifies that the switching function has terminated a routeing server registration.	369
20.1.3 Route Register Cancel	Unregisters the computing function as a routeing server.	370

20.1.1 Route Register

C → S

The Route Register service is used to register the computing function as a routeing server for a specific routeing device or as a routeing server for all routeing devices within the switching sub-domain. The computing function may be required to register for routeing services before it can receive any route requests for a routeing device from the switching function. A computing function may register to be the routeing server for more than one routeing device.

20.1.1.1 Service Request

Table 20-2 Route Register—Service Request

Parameter Name	Type	M/ O/C	Description
routeingDevice	DeviceID	C	Specifies the routeing device for which the computing function requests to be the routeing server. This parameter is mandatory if the switching function does not support the option of registering for all routeing devices in the switching sub-domain. Otherwise, the parameter is optional and if not present, indicates the registration is to be for all routeing devices in the switching sub-domain. See Functional Requirement #2.
requestedRouteingMediaClass	Bitmap	O	Specifies the media classes of calls that are being requested to be routed. Refer to the mediaClass component in 12.2.15, “MediaCallCharacteristics”, on page 113 for the complete set of possible values. Note that multiple bits may be set. If this parameter is not provided (or if the parameter is not supported by the switching function), it is switching function dependent which types of media calls are routed.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

20.1.1.2 Service Response

This service follows the atomic acknowledgement model for this service request.

20.1.1.2.1 Positive Acknowledgement

Table 20-3 Route Register—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
routeRegisterReqID	RouteRegisterReqID	M	Specifies the routeing registration request identifier for this registration.
actualRouteingMediaClass	Bitmap	C	This parameter specifies the actual media classes of calls that are routed by the switching function for routeing registration. The actual media classes of calls routed may be the same or a subset of what was requested on the service request. If this parameter is supported by the switching function, it may be omitted if the requested and actual routeing media class parameters are the same otherwise it shall be provided. If the parameter is not supported by the switching function, then the switching function does not filter media classes for calls for specific routeing registrations. The capability exchange services indicates the media classes of calls that can be routed.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

20.1.1.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

20.1.1.3 Operational Model

20.1.1.3.1 Connection State Transitions

There are no connection state changes due to this service.

20.1.1.3.2 Device-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.1.1.3.3 Call-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.1.1.3.4 Functional Requirements

1. The routeRegisterReqID parameter returned in the positive acknowledgement is used to identify the registration over which routeing requests will be sent. All routeing dialogues for the routeing device occur over this routeing registration. The routeRegisterReqID is also used when cancelling the routeing registration.

2. If the routeingDevice parameter on the service request is not provided and this option is supported, then the registration request is for all routeing devices within the switching sub-domain. If the switching function sends a positive acknowledgement to this request, the routeing server will receive route requests generated for all routeing devices within the switching sub-domain. Some switching function implementations may not support the capability to register for all routeing devices with a single Route Register request (i.e., with the routeingDevice parameter not provided), in which case the switching function will send a negative acknowledgement to the Route Register request. The capabilities exchange services can be used by the computing function to determine if registering for all routeing devices within the switching sub-domain is supported.

3. The number of simultaneous registrations allowed for the same routeing device is switching function dependent. Some switching functions may limit this number to one, in which case only one registration per routeing device is allowed. When the limit is reached, subsequent route register requests for the same routeing device will result in negative acknowledgements from the switching function.

20.1.2 Route Register Abort

S → C

This service is used by the switching function to asynchronously cancel an active routeing registration. This service invalidates a current routeing registration. There is no positive acknowledgement defined for this service.

20.1.2.1 Service Request

Table 20-4 Route Register Abort—Service Request

Parameter Name	Type	M/ O/C	Description
routeRegisterReqID	RouteRegisterReqID	M	Specifies the routeing registration request identifier for the routeing registration that was aborted.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

20.1.2.2 Service Response

There are no service completion conditions for this service.

20.1.2.2.1 Positive Acknowledgement

There is no positive acknowledgement associated with this service.

20.1.2.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

20.1.2.3 Operational Model

20.1.2.3.1 Connection State Transitions

There are no connection state changes due to this service.

20.1.2.3.2 Device-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.1.2.3.3 Call-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.1.2.3.4 Functional Requirements

1. The switching function may issue this service at any time when it can no longer maintain the routeing registration (e.g., when the associated routeing device goes out of service).

20.1.3 Route Register Cancel

C ⟶ S

The Route Register Cancel service is used to cancel a previous route registration. This request terminates the routeing registration and the computing function receives no further routeing requests for that routeing registration once it receives the positive acknowledgement to the Route Register Cancel request.

20.1.3.1 Service Request

Table 20-5 Route Register Cancel—Service Request

Parameter Name	Type	M/ O/C	Description
routeRegisterReqID	RouteRegisterReqID	M	Specifies the routeing registration request identifier for which the routeing registration is to be cancelled.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

20.1.3.2 Service Response

This service follows the atomic acknowledgement model for this service request.

20.1.3.2.1 Positive Acknowledgement

Table 20-6 Route Register Cancel—Positive Acknowledgement

Parameter Name	Type	M/ O/C	Description
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

20.1.3.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

20.1.3.3 Operational Model

20.1.3.3.1 Connection State Transitions

There are no connection state changes due to this service.

20.1.3.3.2 Device-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.1.3.3.3 Call-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.1.3.3.4 Functional Requirements

1. The computing function shall continue to process outstanding routeing requests from the routeing device until it receives a positive acknowledgement for the Route Register Cancel service request. The switching function will not send any further route requests for a registration once it has sent the positive acknowledgement.

20.2 Services

Table 20-7 Call Routeing Services Summary

Routeing Services	Description	Pg.
20.2.1 Re-Route	This service requests an alternate destination from the one provided by a previous Route Select service and based on previous information provided for the call.	372
20.2.2 Route End	This service ends a routeing dialogue.	373
20.2.3 Route Reject	This service is sent to the switching function during a routeing dialogue to indicate that a call should be returned to the network for alternate routeing.	375
20.2.4 Route Request	This service requests that the computing function provides a destination for a call. To aid in the selection of a destination, the service request includes the current destination and may include additional information.	377
20.2.5 Route Select	This service is used by the computing function to provide the destination requested by a previous Route Request or Re-Route request.	379
20.2.6 Route Used	This service provides the actual destination for a call that has been routed using the Route Select service with its optional parameter that requests the route that was used.	381

20.2.1 Re-Route

S → C

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

20.2.1.1 Service Request

Table 20-8 Re-Route—Service Request

Parameter Name	Type	M/ O/C	Description
crossRefIdentifier	RouteingCrossRefID	M	Specifies the cross reference identifier associated with the routeing dialogue.
routeRegisterReqID	RouteRegisterReqID	C	Specifies the route register request identifier associated with the route registration for this routeing dialogue. See Functional Requirement #2.
replyTimeout	Value	O	Specifies the amount of time (in milliseconds) that the switching function will wait for a reply from the computing function, before it proceeds with default routeing for the call. If the parameter is not present or the value is 0, the amount of time is switching function specific.
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call (and if the parameter is supported).
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

20.2.1.2 Service Response

There are no service completion conditions for this service.

20.2.1.2.1 Positive Acknowledgement

There is no positive acknowledgement associated with this service.

20.2.1.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

20.2.1.3 Operational Model

20.2.1.3.1 Connection State Transitions

There are no connection state changes due to this service.

20.2.1.3.2 Device-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.2.1.3.3 Call-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.2.1.3.4 Functional Requirements

1. The requested route is sent from the computing function to the switching function by the Route Select service. The switching function shall use the routeingCrossRefID and routeRegisterReqID (if supported) parameters from the initial Route Request service to link this service to the others that are used to provide a route.
2. The routeRegisterReqID parameter is mandatory if the switching function supports route registration, and shall not be provided otherwise.
3. If the number of remaining retries (remainRetries parameter) is not provided by the computing function in the Route Select service, the computing function should be prepared to respond to all Re-Route service requests or terminate the routeing dialogue by using the Route End service when it cannot provide additional destinations.

20.2.2 Route End

C ◄► S

The Route End service ends a routeing dialogue. This service is bi-directional (i.e., it may be invoked by the switching function or the computing function).

The computing function can use the Route End service when it cannot provide a route for a call. Typically, this can occur if:

- The computing function receives a valid routeing request for a call without sufficient call information and it cannot determine a routeing destination.
- The computing function has already provided all available destinations for a call and no more alternate destinations are available.
- The computing function does not have access to a database necessary to route the call.

In these cases, the computing function uses the Route End service to inform the switching function that it cannot provide a route for the call in question. The Route End service request will terminate the routeing dialogue (routeingCrossRefID) for the call. The Route End request does not clear the call. The switching function will continue to process the call using whatever default routeing algorithm is available (i.e., in a switching function specific way).

The switching function uses the Route End service when it ends a routeing dialogue. Typically, this can occur if:

- The call associated with the routeing cross reference identifier has been successfully routed. This may occur when the computing function has sent a Route Select service request and the switching function has successfully routed the call.
- The calling party has abandoned a call associated with the routeing cross reference identifier.
- The switching function timeout for a route request response has expired. This may occur if the computing function did not respond to a Route Request or Re-Route Request service within a switching function defined period.
- The switching function has ended a routeing dialogue due to internal resource (or other) problems.

20.2.2.1 Service Request

Table 20-9 Route End—Service Request

Parameter Name	Type	M/ O/C	Description
crossRefIdentifier	RouteingCrossRefID	M	Specifies the cross reference identifier associated with the routeing dialogue.
routeRegisterReqID	RouteRegisterReqID	C	Specifies the route register request identifier associated with the route registration for this routeing dialogue. See Functional Requirement #3.
errorValue	ErrorValue	O	Specifies the reason for the route end request (see 12.2.12, “ErrorValue”, on page 88 for more information on this parameter).
correlatorData	CorrelatorData	C	Specifies correlator data. See Functional Requirement #4.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

20.2.2.2 Service Response

There are no service completion conditions for this service.

20.2.2.2.1 Positive Acknowledgement

There is no positive acknowledgement associated with this service.

20.2.2.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

20.2.2.3 Operational Model

20.2.2.3.1 Connection State Transitions

There are no connection state changes due to this service.

20.2.2.3.2 Device-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.2.2.3.3 Call-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.2.2.3.4 Functional Requirements

1. The computing function can use this service to respond to either a Route Request or a Re-Route service request.
2. The routeingCrossRefID and routeRegisterReqID (if supported) parameters from the initial Route Request service shall be used to link this service to the others that are used to provide a route.
3. The routeRegisterReqID parameter is mandatory if the switching function supports route registration, and shall not be provided otherwise.
4. The correlatorData parameter meaning is dependent upon the direction of the service request.
   - if the Route End service request is sent from the Switching Function to the Computing Function then the parameter specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call (and if the parameter is supported).
   - if the Route End service request is sent from the Computing Function to the Switching Function then the parameter specifies the correlator data to associate with the call.

20.2.3 Route Reject

C → S

The Route Reject service request is sent to the switching function during a routeing dialogue to indicate that a call should be returned to the originating network (the network from where the call entered the switching sub-domain where the routeing request was issued from) for alternate routeing.

20.2.3.1 Service Request

Table 20-10 Route Reject—Service Request

Parameter Name	Type	M/ O/C	Description
crossRefIdentifier	RouteingCrossRefID	M	Specifies the cross reference identifier associated with the routeing dialogue.
routeRegisterReqID	RouteRegisterReqID	C	Specifies the route register request identifier associated with the route registration for this routeing dialogue. See Functional Requirement #3.
rejectCause	Enumerated	O	Specifies the reason why call should be returned to the originating network for alternate routeing. The complete set of possible values is: • BusyOverflow - all possible destinations for the call are busy or unavailable and there is no queueing mechanism (ACD, for example) available. • QueueTimeOverflow - all possible destinations for the call are busy or unavailable and the estimated holding time before an agent is able to answer the call is too long. • CapacityOverflow - all possible destinations are busy or unavailable and the call cannot be queued because the system is already at capacity. • CalanderOverflow - all possible calendar based destinations for the call are unavailable because of the time of the day or the day of the week. • UnknownOverflow - the computing function is unable to be more specific. See Functional Requirement #4.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

20.2.3.2 Service Response

There are no service completion conditions for this service.

20.2.3.2.1 Positive Acknowledgement

There is no positive acknowledgement associated with this service.

20.2.3.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

20.2.3.3 Operational Model

20.2.3.3.1 Connection State Transitions

There are no connection state changes due to this service.

20.2.3.3.2 Device-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.2.3.3.3 Call-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.2.3.3.4 Functional Requirements

1. The computing function should issue the Route Reject service only in response to a Route Request or Re-Route Request from the switching function.

2. The routeingCrossRefID and routeRegisterReqID (if supported) parameters from the initial Route Request service shall be used to link this service to the others that are used to provide a route.
3. The routeRegisterReqID parameter is mandatory if the switching function supports route registration, and shall not be provided otherwise.
4. The rejectCause parameter provides additional information why the computing function is requesting that the switching function return the call to the originating network (the network from where the call entered the switching sub-domain where the routeing request was issued from) for alternate routeing. This information can be used by the switching function and/or passed to the originating network via network signalling protocols.

20.2.4 Route Request

S → C

The Route Request service requests that the computing function provide a destination for a call. To aid in the selection of a destination, the service request includes the current destination and may include additional information.

20.2.4.1 Service Request

Table 20-11 Route Request—Service Request

Parameter Name	Type	M/ O/C	Description
crossRefIdentifier	RouteingCrossRefID	M	Specifies the cross reference identifier associated with the routeing dialogue.
routeRegisterReqID	RouteRegisterReqID	C	Specifies the route register request identifier associated with the route registration for this routeing dialogue. See Functional Requirement #3.
currentRoute	CalledDeviceID	M	Specifies the current destination of the call for which a route is requested.
callingDevice	CallingDeviceID	O	Specifies the originator of the call.
routeingDevice	SubjectDeviceID	O	Specifies the device that initiated the Route Request service.
routedCall	ConnectionID	C	Specifies the ConnectionID of the call. This parameter is mandatory if the route request is call related. If the request is not call-related, then this parameter shall not be provided.
routeSelAlgorithm	Enumerated	O	Specifies the type of routeing algorithm requested. The complete set of possible values is: • ACD • Emergency • Least Cost • Normal • User Defined
associatedCallingDevice	AssociatedCallingDeviceID	C	Specifies the Network Interface Device associated with the calling device if the call is an external incoming call. This parameter is mandatory for all external incoming calls and shall not be provided otherwise.
associatedCalledDevice	AssociatedCalledDeviceID	C	For outgoing external calls, this parameter specifies the Network Interface Device associated with the originally called device. For incoming external calls, this parameter specifies a device within the switching sub-domain associated with the originally called device. This parameter is mandatory for all external outgoing calls and it is optional for external incoming calls.
priority	Boolean	O	Specifies the call priority. This may affect the selection of alternative routes. The complete set of possible values is: • True - Priority call. • False - Non-priority call.
replyTimeout	Value	O	Specifies the amount of time (in milliseconds) that the switching function will wait for a reply from the computing function, before it proceeds with default routeing for the call. If the parameter is not present or the value is 0, the amount of time is switching function specific.
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call (and if the parameter is supported).

Table 20-11 Route Request—Service Request (continued)

Parameter Name	Type	M/ O/C	Description
mediaCallCharacteristics	MediaCallCharacteristics	O	This specifies the media class (voice, digital data, etc.) and characteristics of the call. If this parameter is not present (and the parameter is supported) then the call is a voice call.
callCharacteristics	CallCharacteristics	O	Specifies the high level characteristics (ACD call, Priority call, etc.) associated with the call. See 12.2.4 for complete the set of possible values.
routedCallInfo	ConnectionInformation	O	Specifies the connection information associated with the routedCall connection. If this parameter is not present, then the connection information is switching function specific.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

20.2.4.2 Service Response

There are no service completion conditions for this service.

20.2.4.2.1 Positive Acknowledgement

There is no positive acknowledgement associated with this service.

20.2.4.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

20.2.4.3 Operational Model

20.2.4.3.1 Connection State Transitions

There are no connection state changes due to this service.

20.2.4.3.2 Device-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.2.4.3.3 Call-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.2.4.3.4 Functional Requirements

1. The requested route is sent from the computing function to the switching function by the Route Select service.
2. The switching function generates the routeingCrossRefID to link this service to the others that are used to provide a route.
3. The routeRegisterReqID parameter is mandatory if the switching function supports route registration, and shall not be provided otherwise. If the routeRegisterReqID parameter is not present, then the routeingCrossRefID shall be unique across the entire switching sub-domain.

20.2.5 Route Select

C → S

The Route Select service is used by the computing function to provide the destination requested by a previous Route Request or Re-Route service.

20.2.5.1 Service Request

Table 20-12 Route Select—Service Request

Parameter Name	Type	M/ O/C	Description
CrossRefIdentifier	RouteingCrossRefID	M	Specifies the cross reference identifier associated with the routeing dialogue.
routeRegisterReqID	RouteRegisterReqID	C	Specifies the route register request identifier associated with the route registration for this routeing dialogue. See Functional Requirement #3.
routeSelected	DeviceID	M	Specifies the primary selected destination of the call for which a route was requested.
alternateRoutes	List of DeviceIDs	O	Specifies the list of alternate destinations (in priority order) which are to be used sequentially for routeing the call if the primary selected destination initially (i.e., the routeSelected parameter) or a previous entry in the list is not valid or available.

remainRetries	Choice Structure	O	Specifies either the number of alternative routes remaining or the reason why the computing function is not providing it. This shall consist of one of the following choices:
			• noListAvailable (Boolean) - indicates if the computing function does not maintain a fixed list of alternate routes:
			  • TRUE - the computing function does not maintain a list.
			  • FALSE - the computing function does maintain a list.
			• noCountAvailable (Boolean) - indicates if the computing function does not maintain or cannot provide a count of the remaining routes to try:
			  • TRUE - the computing function does not maintain or cannot provide a count.
			  • FALSE - the computing function does maintain a count.
			• retryCount (Value) - indicates the actual number of alternative routes remaining.
			See Functional Requirement #4 for more information on this parameter.

routeUsedReq	Boolean	O	Specifies whether the switching function should issue the Route Used service when it has accepted a route. • True - Route Used service should be issued. • False - Route Used service should not be issued.
correlatorData	CorrelatorData	O	Specifies the correlator data to associate with the call.
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

20.2.5.2 Service Response

There are no service completion conditions for this service.

20.2.5.2.1 Positive Acknowledgement

There is no positive acknowledgement associated with this service.

20.2.5.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

20.2.5.3 Operational Model

20.2.5.3.1 Connection State Transitions

There are no connection state changes due to this service.

20.2.5.3.2 Device-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.2.5.3.3 Call-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.2.5.3.4 Functional Requirements

1. The computing function shall issue a Route Select service request only in response to a Route request or Re-Route request from the switching function.
2. The routeingCrossRefID and routeRegisterReqID (if supported) parameters from the initial Route Request service shall be used to link this service to the others that are used to provide a route.
3. The routeRegisterReqID parameter is mandatory if the switching function supports route registration, and shall not be provided otherwise.
4. If the computing function is unable to supply an actual count (retryCount) of the number of alternative routes in the remainRetries parameter, it can provide one of two other options in the parameter:
   - noListAvailable. This indicates a specific reason why the retry count is not being provided - that the computing function does not maintain a fixed list of alternate routes. For example, the alternative routes may be calculated algorithmically or by progressive database searches (i.e. returns the next entry matched to the search criterion).
   - noCountAvailable. This indicates a more general reason why the retry count is not being provided - that the computing function does not maintain or cannot provide a count of the remaining routes to try. For example, the design of the database does not enable a count of the entries left to be provided or it does not maintain a fixed list of routes.

20.2.6 Route Used

S → C

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

20.2.6.1 Service Request

Table 20-13 Route Used—Service Request

Parameter Name	Type	M/ O/C	Description
crossRefIdentifier	RouteingCrossRefID	M	Specifies the cross reference identifier associated with the routeing dialogue.
routeRegisterReqID	RouteRegisterReqID	C	Specifies the route register request identifier associated with the route registration for this routeing dialogue. See Functional Requirement #4.
routeUsed	CalledDeviceID	M	Specifies the actual destination of the call for which a route was requested.
callingDevice	CallingDeviceID	O	Specifies the originator of the call.
domain	Boolean	O	Specifies whether the resolved destination is within the switching sub-domain. The complete set of possible values is: • True - Resolved destination is within the switching sub-domain. • False - The call has been routed outside the switching sub-domain.
correlatorData	CorrelatorData	C	Specifies the correlator data associated with the call. This parameter is mandatory if there is correlator data associated with the call (and if the parameter is supported).
security	CSTASecurityData	O	Specifies timestamp information, message sequence number, and security information.
privateData	CSTAPrivateData	O	Specifies non-standardized information.

20.2.6.2 Service Response

There are no service completion conditions for this service.

20.2.6.2.1 Positive Acknowledgement

There is no positive acknowledgement associated with this service.

20.2.6.2.2 Negative Acknowledgement

The negative acknowledgement error codes are described in 12.2.12, “ErrorValue”, on page 88.

20.2.6.3 Operational Model

20.2.6.3.1 Connection State Transitions

There are no connection state changes due to this service.

20.2.6.3.2 Device-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.2.6.3.3 Call-Type Monitoring Event Sequences

There are no events generated as a result of this service.

20.2.6.3.4 Functional Requirements

1. This service is used to inform the computing function of the actual route that was used by the switching function. This route could be different than the route provided by the computing function with the Route Select service because the route could have been altered via interactions in the switching function with features such as Forwarding or ACD routeing, for example.
2. The Route Used service should be completed via a Route End service sent by either the computing or switching functions.
3. The routeingCrossRefID and routeRegisterReqID (if supported) parameters from the initial Route Request service shall be used to link this service to the others that are used to provide a route.

4. The routeRegisterReqID parameter is mandatory if the switching function supports route registration, and shall not be provided otherwise.

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