Voice Microservices Events and Models Reference

Basic call models

10/1/2024

Contents

This page describes the basic scenarios in which calls arrive in a contact center.

Related documentation:

-
-
-

RSS:

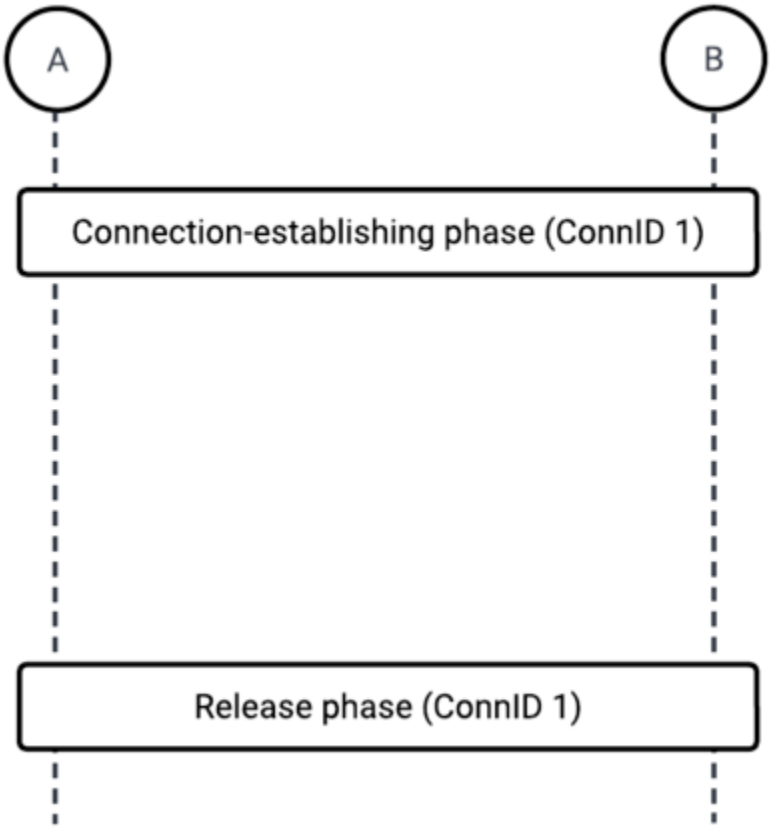
- For private edition

For simplicity, the examples on this page use abbreviated attribute values. For example, ConnID **1**, which – in actual events – displays as ConnID>@metainformation>.

The following comments and abbreviations are used in the call models:

- OPT—Optional.
- DIAL—Might be a dialed number or is not present if Voice Microservices have no information about the other party.

Simple call model



Connection-establishing phase for an internal/inbound call

The following table describes the connection-establishing phase for an internal/inbound call.

Party A	Party B
Make call to B (TMakeCall)	
EventDialing ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN *DIAL OtherDNRole **Destination** *DIAL	
	EventRinging ConnID **1** ThisDN **B** ThisDNRole **Destination**

	OtherDN **A** OtherDNRole **Origination** CallState **OK**
	Answer (TAnswerCall)
EventEstablished ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** OtherDNRole **Destination**	EventEstablished ConnID **1** ThisDN **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination**
Conversation	

Abnormal call flow

Interruption point	Party A	Party B
*	EventReleased ConnID **1** ThisDN **A** ThisDNRole **Origination** CallState **OK**	
**	EventDestinationBusy ConnID **1** ThisDN **A** ThisDNRole **Origination** CallState [a]	
***	EventReleased ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** *DIAL OtherDNRole **Destination** *DIAL CallState **OK**	EventAbandoned ConnID **1** ThisDN **B** OtherDN **A** CallState **OK**

a. CallState might have values that clarify the reason for the destination being busy, for instance `CallState SitInvalidNum`.

Connection-establishing phase for an internal/inbound call to ACD

The following table describes the connection-establishing phase for an internal/inbound call to ACD.

Party A	Party B (ACD Group)	Party C
Make call to B		

EventDialing ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** *DIAL OtherDNRole **Destination** *DIAL	EventQueued ConnID **1** ThisDN **B** ThisQueue **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination**	
	Diverts call to C	
	EventDiverted ConnID **1** ThisDN **B** ThisQueue **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** ThirdPartyDN **C** *OPT ThirdPartyDNRole **Destination** *OPT	
		EventRinging ConnID **1** ThisDN **C** ThisQueue **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** CallState **OK**
		Answer (TAnswerCall)
EventEstablished ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **C** OtherDNRole **Destination**		EventEstablished ConnID **1** ThisDN **C** ThisQueue **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination**
Conversation		

Abnormal call flow

Interruption point	Party A	Party B	Party C
*	EventReleased ConnID **1** ThisDN **A** OtherDN **B** CallState **OK**	EventAbandoned ConnID **1** ThisDN **B** OtherDN **A** CallState **OK**	
**	EventReleased ConnID **1** ThisDN **A** OtherDN **B** CallState **OK**		
***	EventReleased		EventAbandoned

	ConnID **1** ThisDN **A** OtherDN **C** CallState **OK**		ConnID **1** ThisDN **C** OtherDN **A** CallState **OK**

Connection-establishing phase for an internal/inbound call queued to multiple ACDs

The following table describes the connection-establishing phase for an internal/inbound call queued to multiple ACDs.

Party A	Party B (ACD)	Party C (ACD)	Party D
Make internal/ inbound call to B (ACD)			
EventDialing ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** *DIAL OtherDNRole **Destination** *DIAL	EventQueued ConnID **1** ThisDN **B** ThisQueue **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination**		
		EventQueued ConnID **1** ThisDN **C** ThisQueue **C** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination**	
	Diverts call to D		
	EventDiverted ConnID **1** ThisDN **B** ThisDNRole **Origination** OtherDN **C** OtherDNRole **Destination**	EventDiverted ConnID **1** ThisDN **C** ThisQueue **C** ThirdPartyDN **D** ThirdPartyQueue **B** CallState **Redirected** [a]	
			EventRinging ConnID **1** ThisDN **D** ThisQueue **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** CallState **OK**
			Answer (TAnswerCall)

EventEstablished ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **D** OtherDNRole **Destination** CallState **OK**			EventEstablished ConnID **1** ThisDN **D** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** CallState **OK**
Conversation			

a. For ACD configurations where calls are distributed to agents assigned directly to ACD groups, CallState with a value of `Redirected` is present. For ACD configurations where calls are distributed to agents assigned to secondary ACD groups associated with top-level ACD queues, the CallState, with the value `Redirected`, is not present.

Abnormal call flow

Interruption point	Party A	Party B	Party C	Party D
*	EventReleased ConnID **1** ThisDN **A** OtherDN **B** CallState **OK**	EventAbandoned ConnID **1** ThisDN **B** ThisQueue **B** OtherDN **A** CallState **OK**		
**	EventReleased ConnID **1** ThisDN **A** OtherDN **B** CallState **OK**	EventAbandoned ConnID **1** ThisDN **B** ThisQueue **B** OtherDN **A** CallState **OK**	EventAbandoned ConnID **1** ThisDN **C** ThisQueue **C** OtherDN **A** CallState **OK**	
***	EventReleased ConnID **1** ThisDN **A** OtherDN **D** CallState **OK**			
****	EventReleased ConnID **1** ThisDN **A** OtherDN **D** CallState **OK**			EventAbandoned ConnID **1** ThisDN **D** ThisQueue **C** OtherDN **A** CallState **OK**

Connection-establishing phase for an internal/inbound call with call parking

The following table describes the connection-establishing phase for an internal/inbound call with call

parking.

Party A	Party B
Make call to B (TMakeCall)	
EventDialing ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** *DIAL OtherDNRole **Destination** *DIAL	
	Call is parked on B
EventDestinationBusy *OPT ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** *DIAL OtherDNRole **Destination** *DIAL	EventQueued ConnID **1** ThisDN **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** CallState **OK**
	Call is picked up by B
	EventRinging ConnID **1** ThisDN **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** CallState **OK**
	Answer (TAnswerCall)
EventEstablished ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** OtherDNRole **Destination**	EventEstablished ConnID **1** ThisDN **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination**
Conversation	

Abnormal call flow

Interruption point	Party A	Party B
*	EventReleased ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** *DIAL OtherDNRole **Destination** *DIAL CallState **OK**	EventAbandoned ConnID **1** ThisDN **B** OtherDN **A** CallState **OK**

Connection-establishing phase for internal/inbound call with routing (`RouteQueue` case)

The following table describes the connection-establishing phase for an internal/inbound call with routing (`RouteQueue` case).

Party A	Party B (Routing Point/CDN)	Party C
Make incoming call to information service		
EventDialing ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** OtherDNRole **Destination**	EventQueued ConnID **1** ThisDN **B** ThisQueue **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** EventRouteRequest ConnID **1** ThisDN **B** ThisQueue **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination**	
	Route call to C [a] (TRouteCall)	
	EventRouteUsed ConnID **1** ThisDN **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** ThirdPartyDN **C** *OPT ThirdPartyDNRole **Destination** *OPT EventDiverted ConnID **1** ThisDN **B** ThisQueue **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** ThirdPartyDN **C** *OPT ThirdPartyDNRole **Destination** *OPT	
		EventRinging ConnID **1** ThisDN **C** ThisQueue **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** CallState **OK**
		Answer (TAnswerCall)
EventEstablished		EventEstablished

ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **C** OtherDNRole **Destination**		ConnID **1** ThisDN **C** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination**
Conversation		

a. RouteCall to C (TRouteCall()) might be missing.

Abnormal call flow

Interruption point	Party A	Party B	Party C
* and **	EventReleased ConnID **1** ThisDN **A** OtherDN **B** CallState **OK**	EventAbandoned ConnID **1** ThisDN **B** OtherDN **A** CallState **OK**	
***	EventReleased ConnID **1** ThisDN **A** OtherDN **C** CallState **OK**		
****	EventReleased ConnID **1** ThisDN **A** OtherDN **C** CallState **OK**		EventAbandoned ConnID **1** ThisDN **C** OtherDN **A** CallState **OK**

Connection-establishing phase for internal/inbound call with routing

The following table describes the connection-establishing phase for an internal/inbound call with routing.

Party A	Party B (Routing Point/CDN)	Party C
Make incoming call to information service		
EventDialing ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** *DIAL OtherDNRole **Destination** *DIAL	EventRouteRequest ConnID **1** ThisDN **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination**	

	Route call to C [a] (TRouteCall)	
	EventRouteUsed ConnID **1** ThisDN **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** ThirdPartyDN **C** [b] ThirdPartyDNRole **Destination** *OPT CallState **OK/Redirected** [c]	
		EventRinging ConnID **1** ThisDN **C** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** CallState **OK**
		Answer (TAnswerCall)
EventEstablished ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **C** OtherDNRole **Destination**		EventEstablished ConnID **1** ThisDN **C** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination**
Conversation		

a. Not present if a call has been routed by default; that is, a switch did not receive any routing instruction from a computer domain within a timeout configured on the switch side (scripted or otherwise) and therefore processed the call using switch logic.
b. Content of **ThirdPartyDN** depends on the call scenario:

- If information about the destination is available at the moment EventRouteUsed is generated, this attribute is mandatory; a DN where the call has been delivered must be reported.
- If the information is not available, but the call has been routed through Voice Microservices, this attribute is mandatory; a DN where the call has been sent must be reported.
- If a call has been routed to a default destination or routed by another application, this attribute is optional (depends on switch capabilities).

c. **CallState** has a value of `Redirected (22)` if a call has been routed by a switch. For some switches, the attribute **Callstate** might not be present.

Abnormal call flow

Interruption point	Party A	Party B	Party C
*	EventReleased ConnID **1** ThisDN **A** OtherDN **B**	EventAbandoned ConnID **1** ThisDN **B** OtherDN **A**	

	CallState **OK**	CallState **OK**	
**	EventReleased ConnID **1** ThisDN **A** OtherDN **C** CallState **OK**	EventAbandoned [a] ConnID **1** ThisDN **B** OtherDN **A** CallState **OK**	
***	EventReleased ConnID **1** ThisDN **A** OtherDN **C** CallState **OK**		
****	EventReleased ConnID **1** ThisDN **A** OtherDN **C** CallState **OK**		EventAbandoned ConnID **1** ThisDN **C** OtherDN **A** CallState **OK**

a. In this case, EventError must be sent after EventAbandoned to make the ReferenceID available.

Connection-establishing phase for an internal/inbound call with routing outbound

The following table describes the connection-establishing phase for an internal/inbound call with routing outbound.

Party A	**Party B (Routing Point)**	**Party C**
Incoming call		
EventDialing ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** *DIAL OtherDNRole **Destination** *DIAL	EventRouteRequest ConnID **1** ThisDN **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination**	
	Route call to C [a] (TRouteCall)	
EventNetworkReached ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **C** *DIAL OtherDNRole **Destination** *DIAL	EventRouteUsed ConnID **1** ThisDN **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** ThirdPartyDN **C** [b] ThirdPartyDNRole **Destination** *OPT CallState **OK/Redirected** [c]	EventRinging ConnID **1** ThisDN **C** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** CallState **OK**

		Answer (TAnswerCall)
EventEstablished ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **C** OtherDNRole **Destination**		EventEstablished ConnID **1** ThisDN **C** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination**
Conversation		

a. Not present if a call has been routed by default; that is, a switch did not receive any routing instruction from a computer domain within a timeout configured on the switch side (scripted or otherwise) and therefore processed the call using switch logic.
b. Content of **ThirdPartyDN** depends on the call scenario:

- If information about the destination is available at the moment EventRouteUsed is generated, this attribute is mandatory; a DN where the call has been delivered must be reported.
- If the information is not available, but the call has been routed through Voice Microservices, this attribute is mandatory; a DN where the call has been sent must be reported.
- If a call has been routed to a default destination or routed by another application, this attribute is optional (depends on switch capabilities).

c. **CallState** has a value of `Redirected (22)` if a call has been routed by a switch. For some switches, the attribute **CallState** might not be present.

Abnormal call flow

Interruption point	Party A	Party B	Party C
*	EventReleased ConnID **1** ThisDN **A** OtherDN **B** CallState **OK**	EventAbandoned ConnID **1** ThisDN **B** OtherDN **A** CallState **OK**	
**	EventReleased ConnID **1** ThisDN **A** OtherDN **C** CallState **OK**		EventAbandoned ConnID **1** ThisDN **C** OtherDN **A** CallState **OK**

Connection-establishing phase for an outbound call

The following table describes the connection-establishing phase for an outbound call.

Party A	Party B
Make outside call (TMakeCall)	
EventDialing	

ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** *DIAL OtherDNRole **Destination** *DIAL	
EventNetworkReached [a] ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** *DIAL OtherDNRole **Destination** *DIAL	
	Answer
EventEstablished ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** *OPT OtherDNRole **Destination** *OPT	
Conversation	

a. When a switch does not report network reached, Voice Microservices simulate EventNetworkReached right before distributing EventEstablished.

Abnormal call flow

Interruption point	Party A
*	EventReleased ConnID **1** ThisDN **A** OtherDN **B** CallState **OK**
**	EventDestinationBusy ConnID **1** ThisDN **A** OtherDN **B** CallState [a]
***	EventReleased ConnID **1** ThisDN **A** OtherDN **B** CallState **OK**

a. CallState might have values that clarify the reason for the destination being busy, for instance `CallStateSitInvalidNum`.

Connection-establishing phase while on hold (internal/outbound call)

The following table describes the connection-establishing phase for an internal/outbound call while on hold.

Party A	Party B
Call to B	
EventDialing ConnID **1** ThisDN **A** ThisDNRole **Origination** OtherDN **B** OtherDNRole **Destination** CallState **OK**	EventRinging ConnID **1** ThisDN **B** ThisDNRole **Destination** OtherDN **A** OtherDNRole **Origination** CallState **OK**
Hold	
EventHeld ConnID **1** ThisDN **A** OtherDN **B**	
	Answer
EventEstablished ConnID **1** ThisDN **A** OtherDN **B**	EventEstablished ConnID **1** ThisDN **B** OtherDN **A**
Retrieve	
EventRetrieved ConnID **1** ThisDN **A** OtherDN **B** CallState **OK**	