Communicating call events

This chapter includes the following topics:

Call events

Any telephony device can generate a call. Unified CCX, however, knows only the call that is associated with a Unified CCX route point, a Unified CCX CTI port, and a Unified CCX agent phone device into which an agent is logged.

A call connection is the connection between both a caller and a receiver. If a conference call is made, then there are more than two people connected at the same time. Each phone that is part of the call makes up the connection.

Call events are generated by actions relating to calls and are done on the caller's or callee's telephony devices. For example, when a phone rings, a call delivered event is generated. Every action of the caller and callee can also generate a call event. For example the caller or callee can put a call on hold or transfer the call.

Unified CCX generates call events only for the telephony devices that it monitors. Unified CCX normally monitors triggers and agent devices into which agents are logged. In general, if a call contains a monitored device, call events that are relevant to that monitored device are generated.

Unified CCX does not control telephony devices directly. Currently Unified CCX supports two call control products: Cisco Unified Communications Manager and Cisco Unified Communications Manager Express. The procedure to enable an agent device to be monitored depends on these call control products. On Unified CCM, CTI capability must be enabled on agent devices and lines. Please refer to the Unified CCX documentation for details.

Note Unified CME is not supported in Unified CCX 9.0.

Call events are unsolicited messages. A Unified CCX client can select what call events to receive by setting appropriate masks in the OPEN_REQ message. In general, a bridge-mode client receives call events related to all monitored devices while an agent-mode client can receive call events only related to calls that include the agent device.

The order of all call events is not guaranteed. However the CTI client can expect an order for some events. For example, the CALL_DELIVERED event is always before the CALL_ESTABLISHED event.

Call context data

Call context data is information about a call such as, for example, the caller ID or a caller account number. This data is maintained and updated through call variables in the messages.

Unified CCX maintains a set of 10 call variables for each call. Each variable is capable of storing a null terminated string of up to 39 characters (39 variable characters + null termination character = 40 bytes, STRING[40]). When a call is pre-routed by Unified CCX, these variables are initialized to null strings prior to executing the routing script.

The values of the call variables can be also used by Unified CCX to make routing decisions. In addition, the variables can contain added information about the caller, such as the result of a host database query. While routing a call, the routing script can update one or more of the call variables.

Call variables can also be set by a client that is associated with the call by using the SET_CALL_DATA_ REQ message. For example, a client might want to update call data. Unified CCX changes call context data using steps in a Unified CCX script.

A passed call variable with a null string signifies no value. That is, if all the call variables are passed and call variable 2 contains simply a null for a zero length, it will not cause any change to that call variable.

To "blank out" a call variable, a client can enter a single space in the call variable.

Call-Event message summary descriptions

Call-event messages are unsolicited messages generated by the CTI server when a call event occurs and go only from the server to its clients to inform them of the event.

There are no request or confirmation messages associated with unsolicited messages and the content of most of the call-event messages is event specific.

Table 1: Call-Event messages , on page 3 lists the Unified CCX CTI protocol call-event messages. For each message definition, see Message type definitions

Table 1: Call-Event messages

Message	Cause of Message and Purpose
BEGIN_CALL_EVENT	Every call can be announced to the client with an unsolicited BEGIN_CALL_EVENT message, informing the client that it has just been associated with a new call and providing the initial call context data. Additional call and agent state events are then sent to the client as the call is handled. Finally, an END_CALL_EVENT message is sent to the client when its association with a call is dissolved.
END_CALL_EVENT	An END_CALL_EVENT message is generated when the association between a call and the Unified CCX client is dissolved and no further call-event messages for the call are sent to this client.
	Note While this message informs the client that a call has been completely removed from Unified CCX, the message does not necessarily indicate that the subject call has been terminated. For example, the Cisco Unified Communications Manager might have forwarded the call to another call center.
CALL_DATA_UPDATE_EVENT	Changes to the call context data generate a CALL_DATA_UPDATE_EVENT to the client that contains only the items that have changed.
	Each call has a set of data (call context data) associated with it. This data could be caller-entered digits or data in a call variable or a call context variable.
CALL_DELIVERED_EVENT	The arrival of a call at a monitored device on the server generates a CALL_DELIVERED_EVENT message to the client. The call is delivered when the phone rings.
CALL_ESTABLISHED_EVENT	The answering of a call at a device on the server creates a new call connection and generates a CALL_ESTABLISHED_EVENT message to the client.
CALL_HELD_EVENT	Placing a call on hold generates a CALL_HELD_EVENT message to the client.
CALL_RETRIEVED_EVENT	Resuming a call previously placed on hold generates a CALL_RETRIEVED_EVENT message to the client.
CALL_CLEARED_EVENT	A CALL_CLEARED_EVENT message is sent to the client when a call is terminated, normally when the last device disconnects from a call.
CALL_CONNECTION_CLEARED_EVENT	When a party drops from a call, a CALL_CONNECTION_CLEARED_EVENT message can be sent to the client.
CALL_ORIGINATED_EVENT	The initiation of a call from the server at any monitored device generates a CALL_ORIGINATED_EVENT to the client.

| Message	Cause of Message and Purpose
CALL_FAILED_EVENT	A CALL_FAILED_EVENT message is usually sent to the client when a call cannot be completed.
CALL_CONFERENCED_EVENT	The joining of calls into a conference call generates a CALL_CONFERENCED_EVENT to the client.
CALL_DIVERTED_EVENT	The moving of a call from one device to another can generate a CALL_DIVERTED_EVENT to the client.
CALL_TRANSFERRED_EVENT	The transfer of a call to another destination causes a CALL_TRANSFERRED_EVENT message.
CALL_SERVICE_INITIATED_ EVENT	The initiation of telecommunications service ("dial tone") at a device generates a CALL_SERVICE_INITIATED_EVENT to the client.
CALL_QUEUED_EVENT	The placing of a call in a queue pending the availability of some resource generates a CALL_QUEUED_EVENT message to the client.
CALL_DEQUEUED_EVENT	The explicit removal of a call from a queue generates a CALL_DEQUEUED_EVENT message to the client.
RTP_STARTED_EVENT (OPTIONAL)	The RTP_STARTED_EVENT message indicates that an RTP (Real-Time Protocol) media stream (voice) has been started. Since there are two media streams for audio media, there are also two RTP Started events, one indicating the input has started (that is, the phone is listening) and the other that the output has started (that is, the outgoing media from the agent phone has begun). Depending on the Unified CCX configuration, Unified CCX may or may not send this messages. For example, Cisco Unified Communications Manager sends this message, but Cisco Unified Communications Manager Express does not send the message.
RTP_STOPPED_EVENT (OPTIONAL)	The RTP_STOPPED_EVENT message indicates that an RTP media has been stopped. Since there are two media streams for audio media, there are also two RTP Stopped events, one indicating the input has stopped (that is, the phone is not listening) and the other that the output has stopped (that is, the outgoing media from the agent phone has stopped). Depending on the Unified CCX configuration, Unified CCX may or may not send this messages. For example, Cisco Unified Communications Manager sends this message, but Cisco Unified Communications Manager Express does not send the message.

The unspecified flow of call-event messages

The relative order of call-event messages and any corresponding agent-state-change event messages is not specified. An agent-state-event message indicating the agent is in the "talking" state, for example, might be sent before or after the corresponding call established event message.

When an event occurs that would conceptually result in two or more event messages being generated at the same time, the client must be prepared to handle the messages arriving in any order. For example, an agent answering an inbound call might generate both a CALL_ESTABLISHED_EVENT and an AGENT_STATE_EVENT message. These can be received by a client in either order, and other event messages can be sent to the client in between.

Figure 1: A typical unsolicited call-event message flow, on page 6 shows a typical unsolicited call-event message flow.

Figure 1: A typical unsolicited call-event message flow
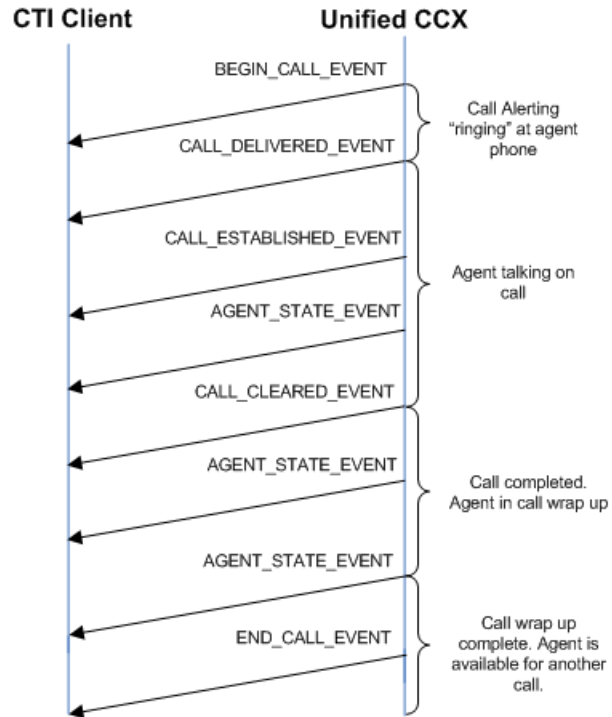
Fields common to all call-event messages

The following fields are in all call-event messages:

- Connection fields that identify a call connection:
  - A numeric **CallID** that identifies a call.
  - A string **ConnectionDeviceID** that identifies each connection of a device to a call.
  - A numeric **ConnectionDeviceType** value that identifies the type of device to which a call is connected.

- An EventCause field that can provide a reason for the occurrence of an event. However, in most cases, no EventCause information is supplied. This is represented by the value CEC_NONE. See Call EventCause (CEC) values, for a list of all the EventCause codes that can be reported.

CallType fields

Some Call-Event messages contain a CallType field that provides the general classification of a call.

Call Types are classified by Unified CCX based on the following scenarios.

CALLTYPE_CONSULT

Whenever a call is originated by placing a consult (even if the call goes off-switch or hits an unmonitored device).

CALLTYPE_AGENT_INSIDE and CALLTYPE_OUT

Whenever an agent places a new call (non-consult), the default is CALLTYPE_AGENT_INSIDE and the call is reclassified to CALLTYPE_OUT if the calls goes off-switch.

CALLTYPE_ACD_IN

All inbound (queued) calls.

CALLTYPE_TRANSFERRED_IN and CALLTYPE_CONFERENCE

All ACD calls (CallType=CALLTYPE_ACD_IN) that are transferred or conferenced.

CALLTYPE_OTHER_IN

Non-ACD-Calls coming to an agent, from outside or from unmonitored devices.

CALLTYPE_ASSIST, CALLTYPE_SUPERVISOR_BARGE_IN, CALLTYPE_SUPERVISOR_INTERCEPT, and CALLTYPE_EMERGENCY

Supervisor Assists, Barge-ins, Intercepts, and Emergency calls.

See CallType values, for a list of all the CallTypes with their meanings.