Cisco TelePresence Conferencing Call Detail Records

File Format Reference Guide

January 2015

Contents

Introduction

Call Detail Records (CDRs) are generated by certain Cisco TelePresence products to provide organizations with historical call data which they can use for billing, auditing, and troubleshooting purposes.

This document describes the CDRs generated by the Cisco TelePresence products listed below. For each product the document lists the event types that can trigger CDRs and describes each event, including an example XML record. It also details the file format of the log files which contain the CDRs.

Table 1: Conferencing products and model numbers

Product	Model numbers
Advanced Media Gateway	AM GW 3610
IP Gateway	IP GW 3500 Series, IP GW MSE 8350
ISDN Gateway	ISDN GW 3241, ISDN GW MSE 8321
MCU	MCU 4200 Series, MCU 4500 Series, MCU 5300 Series, MCU MSE Series
Serial Gateway	Serial GW 3340, Serial GW MSE 8330
TelePresence Server	TelePresence Server on Media 310/320, TelePresence Server on Virtual Machine, TelePresence Server 7010, and TelePresence Server MSE 8710

About the CDR log

The CDR log is stored in memory or on the compact flash card of the device. The log is stored in a proprietary Cisco format which can only be read on a Cisco device. You can download the complete CDR log, or part of it, in XML format using the web interface. The exported log includes all record types and all available details, regardless of the current filtering and display settings in the web interface.

This document explains the format of the log as exported in XML. The file name of the exported data is always **cdr_log.xml**.

CDR file format

CDR files begin with the **`<cdr_events>`** opening tag and close with the matching **`</cdr_events>`** closing tag. For example:

```
<cdr_events>
  <event event_attributes="event values">
    <event_subnodes>
  </event>
  <event event_attributes="event values">
    <event_subnodes>
  </event>
</cdr_events>
```

Event nodes

All **`event`** nodes have the same attributes. The attribute values help to uniquely distinguish the events. For example:

```
<event index="21765553" date="17 April 2011" time="16:02:48" type="new_connection">
```

Table 2: Event node common attributes

Event attribute	Attribute description
index	Unique to the unit. An auto-incremented integer that identifies the event.
date	Date of the event in **`dd Month yyyy`** format.
time	Time of the event in **`hh:mm:ss`** format (24 hour clock).
type	The event type.

Timestamps

If the device time is changed (by changing the system time or via an NTP update) then new events in the CDR log will show the new time. Timestamps on existing logged CDR events remain unchanged.

Events that trigger CDRs

When CDR logging is enabled on a Cisco TelePresence product, records are generated for the following events:

Table 3: Triggers for Call Detail Records, by product

Product	Triggers for CDRs
Advanced Media Gateway	When a call starts, completes, or is disconnected for some other reason. See Advanced Media Gateway event types [p.59].
IP Gateway	When a call starts, transfers to a multisite call, or completes or is disconnected for some other reason. See IP Gateway event types [p.48].
ISDN Gateway	When a call starts, transfers to a multisite call, or completes or is disconnected for some other reason. Even if logging is disabled the gateway still generates CDRs (although they are not stored). See ISDN Gateway event types [p.36].
MCU	When a conference starts or finishes, and in response to other events such as participants joining and leaving the conference. See MCU event types [p.6].
Serial Gateway	When a call starts, transfers to a multisite call, or completes or is disconnected for some other reason. Even if logging is disabled the gateway still generates CDRs (although they are not stored). See Serial Gateway event types [p.64].
TelePresence Server	When conferences start or finish, are active or inactive, and when participants connect, join, leave, and disconnect. The CDR also includes a media summary for each participant. See TelePresence Server event types [p.17]. CDR logging is always enabled on the TelePresence Server and cannot be disabled. These devices store the latest 2000 records only, discarding earlier records as necessary. They also do not write logs to compact flash—they hold the records in memory.

MCU event types

The Cisco TelePresence MCU generates the following records:

- scheduled_conference_started [p.7] when a scheduled conference was started. Either a permanent conference or one with a scheduled end time.
- ad-hoc_conference_started [p.8] when an ad hoc conference was started through the auto attendant.
- participant_joined [p.9] when a participant joined the conference.
- participant_left [p.10] when a participant disconnected or was forcibly disconnected.
- conference_finished [p.15] when the conference finished.

Change summary

Table 4: Changes for Cisco TelePresence MCU Version 4.5

Event type	Node	Change
no media received [p.13]	disconnect reason	Introduced

Table 5: Changes for Cisco TelePresence MCU Version 4.4

Event type	Node	Change
participant_left [p.10]	call	Details added for `disconnect_reason` attribute

scheduled_conference_started

There are two variations for this event. One for permanent conferences and one for conferences with a scheduled end time. The differences are indicated in the event reference table.

Example XML

```
<event index="0" date="13 April 2010" time="13:09:14" type="scheduled_conference_starte
d">
  <conference unique_id="3365002" name="Temporary conference" />
  <conference_details numeric_id="1111" has_pin="no" billing_code="&lt;none&gt;" />
  <owner name="admin" />
  <end scheduled_date="13 April 2010" scheduled_time="13:18:00" scheduled_duration_in_min
utes="9" />
</event>
```

Event reference

Table 6: scheduled_conference_started XML reference

Node	Attribute	Description
conference		
	unique_id	Unique identifier for the conference in the format **nnnnnnnn**. This is generated automatically by the MCU.
	name	For scheduled conferences, this is the conference name as allocated by the user. For ad hoc conferences, it is a name provided by the MCU.
conference_ details		
	numeric_id	Numeric id given to the conference by the creator or **<none>**. Used either for calling into a conference via a gatekeeper or calling in using the MCU as an H.323 gateway.
	has_pin	Whether or not a PIN was used to enter the conference. This will be either **yes** or **no**. Note that PINs are optional for scheduled conferences.
	billing_code	For future expansion.
owner		
	name	Log in user name of the person who created the conference.
end		
	scheduled_ date	End date of the conference in the format **dd Month yyyy** unless this is a permanent conference in which case the end date is not included.
	scheduled_ time	Either the time in the format **hh:mm:ss** or permanent.
	scheduled_ duration_in_ minutes	Scheduled length of the conference in minutes. Not included for permanent conferences.

ad-hoc_conference_started

This event is logged when a conference is started from the MCU's auto attendant with the **Create new conference** option.

Example XML

```
<event index="2" date="13 April 2010" time="13:10:22" type="ad-hoc_conference_started">
  <conference unique_id="3365005" name="3333" />
  <conference_details numeric_id="3333" has_pin="no" billing_code="&lt;none&gt;" />
  <creator participant_id="1" />
  <end scheduled_time="&lt;none&gt;" />
</event>
```

Event reference

Table 7: ad-hoc_conference_started XML reference

Node	Attribute	Description
conference		
	unique_id	Unique identifier for the conference in the format **nnnnnnnn**. This is generated automatically by the MCU.
	name	Usually the same as the numeric_id.
conference_details		
	numeric_id	The conference ID entered by the creator of the conference or **<none>**.
	has_pin	Whether or not a PIN was used to enter the conference. This will be either **yes** or **no**. Note that PINs are optional for ad hoc conferences.
	billing_code	Reserved for future expansion. Always **<none>**.
creator		
	participant_id	Unique number that identifies the participant who created the conference.
end		
	scheduled_time	Not relevant to an ad hoc conference and therefore always **<none>**.

participant_joined

This event is logged whenever a participant joins a conference.

Example XML

```
<event index="3" date="13 April 2010" time="13:10:26" type="participant_joined">
  <conference unique_id="3365005" name="3333" />
  <participant particpant_id="1" participant_id="1" />
  <call direction="incoming" />
</event>
```

Event reference

Table 8: participant_joined XML reference

Node	Attribute	Description
conference		
	unique_id	Unique identifier in the format nnnnnnn for the conference seen in the **scheduled_conference_started** or **ad-hoc_conference_started** events.
	name	For scheduled conferences, the conference name as allocated by the user and, for ad hoc conferences, a name allocated by the unit.
participant *		
	participant_id	Unique number in the format **nnnnnnn** for this participant, automatically generated by the MCU.
call		
	direction	Either **incoming** or **outgoing**.

* Within this event you will see a **particpant_id** and a **participant_id** attribute in the **participant** node because of the need to correct a spelling mistake in the code.

participant_left

This event is logged whenever a participant leaves a conference.

Example XML

```
<event index="4" date="13 April 2010" time="13:12:41" type="participant_left">
  <conference unique_id="3365005" name="3333" />
  <endpoint_details ip_address="10.2.160.3" dn="&lt;none&gt;" h323_alias="sam.spade.e20o1
@cisco.com" configured_name="&lt;none&gt;" />
  <participant participant_id="1" />
  <call time_in_conference="2 mins 15 sec" time_in_conference_in_minutes="3" disconnect_r
eason="participant ended call" />
  <media_from_endpoint resolution="1280 x 768" video_codec="H.264" audio_codec="AAC" band
width="832000 bit/s" />
  <media_to_endpoint resolution="768 x 512" video_codec="H.264" audio_codec="AAC" bandwid
th="832000 bit/s" />
</event>
```

Event reference

Table 9: participant_left XML reference

Node	Attribute	Description
conference		
	unique_id	Unique identifier in the format **nnnnnnnn** for the conference seen in the **scheduled_conference_started** or **ad-hoc_conference_started** events.
	name	For scheduled conferences, it is the conference name as allocated by the user. For ad hoc conferences, it is a name allocated by the unit.
endpoint_details		
	ip_address	IP address of the endpoint.
	dn	E.164 number of the endpoint.
	h323_alias	Configured endpoint name.
	configured_name	Name of endpoint as it appears in the **Endpoints** page on the MCU web interface.
participant		
	participant_id	Unique number (n or nn) for this participant, as generated by the MCU when the participant joined the conference.
call		
	time_in_conference	Duration that the participant was connected to the conference in minutes and seconds.
	time_in_conference_in_minutes	Duration that the participant was connected to the conference rounded up to the next minute.
	disconnect_reason	A string explaining why the participant was disconnected. See Disconnect reasons [p.12] for details.

Table 9: participant_left XML reference (continued)

Node	Attribute	Description
media_from_endpoint		
	resolution	The highest resolution sent to or received from the endpoint during the course of its conference participation. Resolution is listed in the format **`w x h`**. For example, **`704 x 576`**.
	video_codec	One of: ■ Null ■ H.261 ■ Motion ■ JPEG ■ MPEG2 system stream raw ■ H.263 ■ H.264 ■ Remote frame buffer
	audio_codec	One of: ■ Null ■ G.711a ■ G.711mu ■ MPEG2 system stream raw ■ Linear ■ G.711mu ASF ■ G.722 ■ G.722.1 ■ G.722.1 Annex C ■ G.723.1 ■ G.728 ■ G.729 ■ G.729A ■ G.729B ■ G.729AB ■ Polycom(R) Siren14(TM) ■ AAC
	bandwidth	Bandwidth in bits per second.
media_to_endpoint		
	resolution	The highest resolution sent to or received from the endpoint during the course of its conference participation. Resolution is listed in the format **`w x h`**. For example, **`704 x 576`**.
	video_codec	As for **`media_from_endpoint`** above.
	audio_codec	As for **`media_from_endpoint`** above.
	bandwidth	Bandwidth in bits per second.

Disconnect reasons

Table 10: Explanations for the possible disconnect reasons

Disconnect reason	Explanation
`all participants dropped`	The MCU disconnected all participants from the conference. This could be the result of a scheduled conference ending, a web user deliberately disconnecting all participants, or an API call ending the conference.
`failed to authenticate with vnc server`	The MCU and the endpoint could not authenticate each other when trying to establish a secure connection.
`busy`	The MCU could not make the connection because the endpoint was on another call.
`capset error`	The capability set from the MCU was rejected, or the MCU did not receive a reply to its capability message. Check your endpoint is running the latest version, and that there is no network congestion that could stop messages reaching the MCU.
`conference doesn't support ConferenceMe`	A ConferenceMe participant is trying to join a conference when ConferenceMe is disabled either in the conference settings or the global streaming settings.
`destination unreachable`	The MCU cannot establish the call because it cannot reach the remote endpoint. The endpoint may be switched off, the IP address may be incorrect, or the destination may be incapable of receiving a call.
`DNS failure`	The address typed was not registered to a gatekeeper, could not be dialed as an IP address and could not be found with a DNS lookup.
`failed to connect to vnc server`	Unable to connect to VNC server. This can be due to a network problem or if a VNC server is not listening on the specified host.
`gatekeeper required`	The MCU settings require that a gatekeeper be present, but the gatekeeper is not responding.
`H.225 decode error`	The MCU was unable to decode an incoming H.225 message.
`H.225 protocol error`	There has been an H.225 protocol error. For example the endpoint has sent an invalid H.225 message to the MCU.
`H.225 socket error`	There has been an error establishing a TCP connection to the H.225 socket on the endpoint. For example there is no route to the desired IP address.
`H.245 decode error`	The MCU was unable to decode the incoming H.245 message.
`H.245 protocol error`	There has been an H.245 protocol error. For example the endpoint has sent an invalid H.245 message to the MCU.
`H.245 socket error`	There has been an error establishing a TCP connection to the H.245 socket on the endpoint. For example the endpoint is not listening on the H.245 port it had previously specified.

Table 10: Explanations for the possible disconnect reasons (continued)

`incompatible vnc version`	VNC version is incompatible with MCU. See Using VNC with Cisco TelePresence MCU for details of supported versions.
`local gatekeeper refused`	The gatekeeper to which the MCU is registered refused to complete the call. This may occur if the gatekeeper cannot route the call or blocks it for security reasons.
`internal overflow`	An excess of information in the message buffer has caused it to run out of space and overflow.
`moved`	The participant was moved to another conference.
`network error`	There has been an unspecified network error.
`no answer`	The endpoint started ringing but the call was not accepted by the user.
`no conference for ConferenceMe`	A ConferenceMe user disconnected because ConferenceMe could not find a conference with ConferenceMe enabled.
`no gatekeeper`	The address could not be resolved as an IP address, but no gatekeeper is set on the **Settings > Gatekeeper page** to resolve the number into an E.164 address.
`no media received`	No media packets were received from a participant for 30 seconds without hold or mute having been signaled.
`participant dropped`	The MCU ended the call, for example if a user hung up the call via the web interface.
`participant ended call`	The endpoint hung up a call that was in progress.
`port allocation exceeded`	The MCU could not honour this connection because there were no available ports.
`protocol error`	There has been an unspecified protocol error.
`Q.931 decode error`	The MCU was unable to decode an incoming Q.931 message.
`Q.931 protocol error`	There has been a Q.931 protocol error. For example the endpoint has sent an invalid Q.931 message to the MCU.
`rejected`	The participant chose to reject the incoming call instead of answering.
`rejected immediately`	The endpoint rejected the call without ringing.
`remote gatekeeper refused`	The remote gatekeeper refused the request from the the remote endpoint.
`remote gatekeeper unreachable`	The remote gatekeeper did not respond to the endpoint that the MCU was trying to call.

Table 10: Explanations for the possible disconnect reasons (continued)

`remote gateway resources`	The remote gateway has insufficient resources to let the call complete. For example the call is being routed to an ISDN gateway with insufficient channels to allow the call to complete.
`service unavailable`	The requested service is unavailable. This directly corresponds to an H.323 or SIP message received from the far end to indicate that the call is unable to proceed. The far end could have made this decision for any one of a number of reasons, including lack of resource availability or a call routing policy that prevents the MCU from calling the destination number.
`timeout`	No reply from the endpoint, for example if network problems prevented any messages reaching the endpoint from the MCU, or vice versa.
`unknown`	This disconnect reason should not appear in the CDR record and it may indicate a software fault. Check the event log which may provide more details.
`unspecified`	This disconnect reason should not appear in the CDR record and it may indicate a software fault. Check the event log which may provide more details.
`unspecified error`	This disconnect reason should not appear in the CDR record and it may indicate a software fault. Check the event log which may provide more details.
`video port limit exceeded`	The MCU could not honour this connection because there were no available video ports.

conference_finished

This event is logged when a conference completes according to its schedule end time or is terminated.

Example XML

```
<event index="6" date="13 April 2010" time="13:18:00" type="conference_finished">
  <conference unique_id="3365002" name="Temporary conference" />
  <limits audio_video_participants="20" audio_only_participants="60" streaming_participan
ts_allowed="1" />
  <participants max_simultaneous_audio_video="0" max_simultaneous_audio_only="0" max_simu
ltaneous_streaming="0" total_audio_video="0" total_audio_only="0" total_streaming="0" />
  <gatekeeper registered_with_gatekeeper="no" />
  <end duration="10 mins 0 sec" duration_in_minutes="10" />
</event>
```

Event reference

Table 11: conference_finished XML reference

Node	Attribute	Description
conference		
	unique_id	Unique identifier for the conference in the format **nnnnnnnn** as seen in the **scheduled_conference_started** or **ad-hoc_conference_started** events.
	name	For scheduled conferences, this is the conference name as allocated by the user. For ad hoc conferences, it is a name provided by the MCU.
limits		
	audio_video_participants	The maximum number of video plus audio participants that were allowed on this conference. This limit can either be explicitly set by the conference owner or will be the maximum number of participants that the MCU supports.
	audio_only_participants	The maximum number of audio-only participants that were allowed on this conference. This limit can either be explicitly set by the conference owner or will be the maximum number of participants that the MCU supports.
	streaming_participants_allowed	Either `1` (allowed) or `0` (not allowed).

Table 11: conference_finished XML reference (continued)

Node	Attribute	Description
participants		
	max_ simultaneous_ audio_video	The highest number of a type of participant present at any one time during the lifetime of the conference.
	max_ simultaneous_ audio_only	
	max_ simultaneous_ streaming	
	total_audio_ video	The total number of a type of participant who joined the conference during its lifetime.
	total_audio_ only	
	total_ streaming	
gatekeeper		
	registered_ with_ gatekeeper	The value is `yes` if the conference was ever registered with a gatekeeper. The value is `no` if the conference was never registered with a gatekeeper.
end		
	duration	How long the conference lasted in minutes and seconds.
	duration_in_ minutes	How long the conference lasted rounded up to the next whole number of minutes.

TelePresence Server event types

The Cisco TelePresence Server generates the following records:

- conference_started [p.19] when a conference starts.
- conference_finished [p.21] when a conference ends.
- conference_active [p.22] when the first participant joins an inactive conference.
- conference_inactive [p.23] when the last participant leaves an active conference.
- participant_connected [p.25] when a participant connects to the TelePresence Server.
- participant_disconnected [p.27] when a participant disconnects from the TelePresence Server.
- participant_joined [p.29] when a participant joins a conference.
- participant_left [p.30] when a participant leaves a conference.
- participant_media_summary [p.31] when the TelePresence Server saves the media statistics of the call.

Change summary

Table 12: Changes for TelePresence Server 4.1

Event type	Node	Change
participant_media_summary [p.31]	packets_lost	Nodes amended to include rx and tx streams.
participant_media_summary [p.31]	frames_sent, frames_received, frame_errors, furs_sent, furs_received	Nodes added.

Table 13: Changes for TelePresence Server 4.0

Event type	Node	Change
participant_disconnected [p.27]	call_direction, call_protocol, endpoint_ip_address, endpoint_display_name, endpoint_uri, endpoint_configured_name	Nodes added to correspond with **`participant_connected`**
participant_connected [p.25]	Nodes reordered	Documentation updated to correspond with **`participant_disconnected`**.

Table 14: Changes for TelePresence Server 3.1

Event type	Node	Change
conference_finished [p.21]	total_audio_video_participants, total_audio_only_participants, max_simultaneous_audio_only_participants	Documentation corrected
conference_inactive [p.23]	total_audio_video_participants, total_audio_only_participants, max_simultaneous_audio_only_participants	Documentation corrected
participant_media_summary [p.31]	stream node and example XML	Documentation corrected

Table 15: Changes for TelePresence Server 3.0

Event type	Node	Change
conference_started [p.19]	billingCode	Addition
participant_disconnected [p.27]	disconnectReason	Removed unused reasons

conference_started

This event is logged when a conference starts.

Example XML

```
<event index="58870" date="11 July 2012" time="09:58:31" type="conference_started">
  <conference_started>
    <conference_guid>0b908c92-f5ba-4591-a45d-5571d0760896</conference_guid>
    <name>My Conference</name>
    <billing_code>my_billing_code</billing_code>
    <numeric_id>971771</numeric_id>
    <uris>
    <uri>
    <uri>971771</uri>
    <pin_protected>yes</pin_protected>
    </uri>
    </uris>
    <scheduled_date>11 July 2012</scheduled_date>
    <scheduled_time>12:15:00</scheduled_time>
  </conference_started>
</event>
```

Event reference

Table 16: conference_started XML reference

Nodes and nesting				Description
conference_started				
	conference_guid			Globally Unique Identifier (GUID) of this conference.
	name			Name of the conference.
	billing_code			User-supplied billing code for this conference. Note that billing_code is only present if one was supplied at the time of conference creation.
	numeric_id			Numeric ID of the conference if available. This is omitted from the record if it is unavailable to the TelePresence Server. (The uri array of structs is the recommended parameter for version 2.3 onwards—numeric_id is retained for version 2.2 backward compatibility.)
	uris			The uri array of structs is the recommended parameter for version 2.3 onwards. (numeric_id is retained for version 2.2 backward compatibility.)
		uri		Each uri contains a **`uri`** and a **`pin-protected`** for the conference.
			uri	A uri for the conference.
			pin_protected	Whether or not a PIN is required to access the conference by this uri. This will be either **`yes`** or **`no`**.
	scheduled_date			Start date of a scheduled conference. This is only present for scheduled conferences. (Not visible when accessed using the API.) The scheduled date format would appear as **`11 July 2012`** for example.
	scheduled_time			Start time of a scheduled conference. This is only present for scheduled conferences. The scheduled time format would appear as **`12:15:00`** for example.

conference_finished

This event is logged when a conference ends.

Example XML

```
<event index="58883" date="11 July 2012" time="10:15:41" type="conference_finished">
  <conference_finished>
    <conference_guid>0b908c92-f5ba-4591-a45d-5571d0760896</conference_guid>
    <max_simultaneous_audio_video_participants>2</max_simultaneous_audio_video_participan
ts>
    <max_simultaneous_audio_only_participants>1</max_simultaneous_audio_only_
participants>
    <total_audio_video_participants>2</total_audio_video_participants>
    <total_audio_only_participants>1</total_audio_only_participants>
    <duration>1030</duration>
  </conference_finished>
</event>
```

Event reference

Table 17: conference_finished XML reference

Nodes and nesting		Description
conference_finished		
	conference_guid	Globally Unique Identifier (GUID) of this conference.
	max_simultaneous_audio_video_participants	Count of the maximum (peak) number of participants who were using audio and video at the same time.
	max_simultaneous_audio_only_participants	Count of the maximum (peak) number of participants who were using audio only at the same time.
	total_audio_video_participants	Total number of unique participants who were using both audio and video at some point during their participation in the conference.
	total_audio_only_participants	Total number of unique participants who were audio-only for the duration of their participation in the conference. **Note:** In the event of an audio-only participant becoming a video participant during the conference, or vice versa, the participant is counted in **`total_audio_video_participants`** and not in **`total_audio_only_participants`**. This means that the total for audio-only participants could be lower than the count of **`max_simultaneous_audio_only_participants`**.
	duration	Total time elapsed, in seconds, since this conference started.

conference_active

This event is logged when the first participant joins an inactive conference. The period between the **conference_active** and corresponding **conference_inactive** events is called a session.

Example XML

```
<event index="58872" date="11 July 2012" time="10:07:03" type="conference_active">
  <conference_active>
    <conference_guid>0b908c92-f5ba-4591-a45d-5571d0760896</conference_guid>
  </conference_active>
</event>
```

Event reference

Table 18: conference_active XML reference

Nodes and nesting		Description
conference_active		
	conference_guid	Globally Unique Identifier (GUID) of this conference.

conference_inactive

The TelePresence Server logs this event when the last participant leaves an active conference. The period between the **conference_active** and corresponding **conference_inactive** events is called a session.

Example XML

```
<event index="58880" date="11 July 2012" time="10:15:30" type="conference_inactive">
  <conference_inactive>
    <conference_guid>0b908c92-f5ba-4591-a45d-5571d0760896</conference_guid>
    <max_simultaneous_audio_video_participants>2</max_simultaneous_audio_video_participan
ts>
    <max_simultaneous_audio_only_participants>0</max_simultaneous_audio_only_
participants>
    <total_audio_video_participants>2</total_audio_video_participants>
    <total_audio_only_participants>1</total_audio_only_participants>
    <session_duration>507</session_duration>
  </conference_inactive>
</event>
```

Event reference

Table 19: conference_inactive XML reference

Nodes and nesting		Description
conference_inactive		
	conference_guid	Globally Unique Identifier (GUID) of this conference.
	max_simultaneous_audio_video_participants	Count of the (peak) maximum number of participants who were using audio and video at the same time during the session.
	max_simultaneous_audio_only_participants	Count of the (peak) maximum number of participants who were using audio only at the same time during the session.
	total_audio_video_participants	Total number of unique participants who were using both audio and video at some point during their participation in the session.
	total_audio_only_participants	Total number of unique participants who were audio-only for the duration of their participation in the session. **Note:** In the event of an audio-only participant becoming a video participant during the session, or vice versa, the participant is counted in **`total_audio_video_participants`** and not in **`total_audio_only_participants`**. This means that the total for audio-only participants could be lower than the count of **`max_simultaneous_audio_only_participants`**.
	session_duration	Period of time, in seconds, for which this conference was active in the session ended at the time of this record.

participant_connected

A participant has connected to the TelePresence Server.

Example XML

```
<event index="58871" date="11 July 2012" time="10:07:03" type="participant_connected">
  <participant_connected>
    <participant_guid>ed9b5670-9dbc-4b6d-8112-4a06e805f2b2</participant_guid>
    <call_id>ed9b5670-9dbc-4b6d-8112-4a06e805f2b2</call_id>
    <call_direction>outgoing</call_direction>
    <call_protocol>sip</call_protocol>
    <endpoint_ip_address>192.168.0.3</endpoint_ip_address>
    <endpoint_display_name>emplname@cisco.com</endpoint_display_name>
    <endpoint_uri>emplname@cisco.com</endpoint_uri>
    <endpoint_configured_name>emplname@cisco.com</endpoint_configured_name>
  </participant_connected>
</event>
```

Event reference

Table 20: **participant_connected** event reference

Nodes and nesting		Description
participant_connected		
	participant_guid	Globally Unique Identifier (GUID) of this participant. This GUID is retained for the duration of the connection.
	call_id	Unique to each constituent call of a participant
	call_direction	■ **outgoing**: The TelePresence Server called this participant. ■ **incoming**: This participant called the TelePresence Server. ■ **unknown**: The TelePresence Server does not know the **call_direction** of the disconnected call.
	call_protocol	■ **h323** ■ **sip** ■ **unknown**
	endpoint_ip_address	Endpoint's IP address if available. This is omitted from the record if it is unavailable to the TelePresence Server.
	endpoint_display_name	Endpoint's display name if present. This is omitted from the record if it is unavailable to the TelePresence Server. **Note:** **endpoint_display_name** can change during a call, so this value could differ from the corresponding value in the **participant_disconnected** record.

Table 20: participant_connected event reference (continued)

Nodes and nesting		Description
	endpoint_uri	For outgoing calls: the call out address. For incoming H.323 calls: the call in address, if available (otherwise the E.164). For incoming SIP calls: the SIP URI.
	endpoint_configured_name	The name of the endpoint as configured on the TelePresence Server (if it has a configured name) otherwise one of its call-in parameters, e.g. its URI. **Note:** **`endpoint_configured_name`** can change during a call, so this value could differ from the corresponding value in the **`participant_disconnected`** record.

participant_disconnected

A participant has disconnected from, or has been disconnected by, the TelePresence Server.

Example XML

```
<event index="58881" date="11 July 2012" time="10:15:30" type="participant_disconnected">
  <participant_disconnected>
    <participant_guid>ed9b5670-9dbc-4b6d-8112-4a06e805f2b2</participant_guid>
    <call_id>ed9b5670-9dbc-4b6d-8112-4a06e805f2b2</call_id>
    <disconnect_reason>remote_teardown</disconnect_reason>
    <call_direction>outgoing</call_direction>
    <call_protocol>sip</call_protocol>
    <endpoint_ip_address>192.168.0.3</endpoint_ip_address>
    <endpoint_display_name>emplname@cisco.com</endpoint_display_name>
    <endpoint_uri>emplname@cisco.com</endpoint_uri>
    <endpoint_configured_name>emplname@cisco.com</endpoint_configured_name>
  </participant_disconnected>
</event>
```

Event reference

Table 21: **`participant_disconnected`** event reference

Nodes and nesting		Description
participant_disconnected		
	participant_guid	Globally Unique Identifier (GUID) of the participant that was disconnected from the TelePresence Server. This GUID is retained for the duration of the connection.
	call_id	Unique to each constituent call of a participant
	disconnect_reason	The reason that the participant was disconnected. ▪ **`unspecified`**: The TelePresence Server does not know why the call disconnected. ▪ **`local_teardown`**: The TelePresence Server disconnected the call. ▪ **`remote_teardown`**: The endpoint disconnected the call.
	call_direction	▪ **`outgoing`**: The TelePresence Server called this participant. ▪ **`incoming`**: This participant called the TelePresence Server. ▪ **`unknown`**: The TelePresence Server does not know the **`call_direction`** of the disconnected call.
	call_protocol	▪ **`h323`** ▪ **`sip`** ▪ **`unknown`**

Table 21: participant_disconnected event reference (continued)

Nodes and nesting		Description
	endpoint_ip_address	Endpoint's IP address if available. This is omitted from the record if it is unavailable to the TelePresence Server.
	endpoint_display_name	Endpoint's display name if present. This is omitted from the record if it is unavailable to the TelePresence Server. **Note:** **`endpoint_display_name`** can change during a call, so this value could differ from the corresponding value in the **`participant_connected`** record.
	endpoint_uri	For outgoing calls: the call out address. For incoming H.323 calls: the call in address, if available (otherwise the E.164). For incoming SIP calls: the SIP URI.
	endpoint_configured_name	The name of the endpoint as configured on the TelePresence Server (if it has a configured name) otherwise one of its call-in parameters, e.g. its URI. **Note:** **`endpoint_configured_name`** can change during a call, so this value could differ from the corresponding value in the **`participant_connected`** record.

participant_joined

A participant has joined the conference.

Example XML

```
<event index="58873" date="11 July 2012" time="10:07:03" type="participant_joined">
  <participant_joined>
    <conference_guid>0b908c92-f5ba-4591-a45d-5571d0760896</conference_guid>
    <participant_guid>ed9b5670-9dbc-4b6d-8112-4a06e805f2b2</participant_guid>
    <call_id>ed9b5670-9dbc-4b6d-8112-4a06e805f2b2</call_id>
  </participant_joined>
</event>
```

Event reference

Table 22: participant_joined XML reference

Nodes and nesting		**Description**
participant_joined		
	conference_guid	Globally Unique Identifier (GUID) of the conference that this participant joined.
	participant_guid	Globally Unique Identifier (GUID) of the participant that joined this conference. This GUID is retained for the duration of the connection.
	call_id	Unique to each constituent call of a participant .

participant_left

A participant has left the conference.

Example XML

```
<event index="58879" date="11 July 2012" time="10:15:30" type="participant_left">
  <participant_left>
    <conference_guid>0b908c92-f5ba-4591-a45d-5571d0760896</conference_guid>
    <participant_guid>ed9b5670-9dbc-4b6d-8112-4a06e805f2b2</participant_guid>
    <call_id>ed9b5670-9dbc-4b6d-8112-4a06e805f2b2</call_id>
    <time_in_conference>506</time_in_conference>
  </participant_left>
</event>
```

Event reference

Table 23: participant_left XML reference

Nodes and nesting		Description
participant_left		
	conference_guid	Globally Unique Identifier (GUID) of the conference that the participant left.
	participant_guid	Globally Unique Identifier (GUID) of the participant that left this conference. This GUID is retained for the duration of the connection
	time_in_conference	Period of time, in seconds, that this participant spent in this conference.
	call_id	Unique to each constituent call of a participant.

participant_media_summary

A summary of the media transfer between the TelePresence Server and the endpoint while the participant is connected to the TelePresence Server. The summary includes information about all the streams between the endpoint and the TelePresence Server.

Each stream node identifies the direction and type of the stream as well as the codecs used and the packet statistics for the stream at the time of the **`participant_media_summary`** record.

Example XML

```
<participant_media_summary>
<participant_guid>5903af70-ba84-11e3-b655-000d7c10d038</participant_guid>
 <call_id>5903af70-ba84-11e3-b656-000d7c10d038</call_id>
 <streams>
  <stream direction="rx" type="video" context="main" position="0" encryption_status="encr
ypted">
   <codecs>
    <codec>
     <name>H.264</name>
     <active_time>1376</active_time>
     <encrypted_time>1376</encrypted_time>
    </codec>
   </codecs>
   <width>1280</width>
   <height>720</height>
   <max_bandwidth>4014</max_bandwidth>
   <bandwidth>3974</bandwidth>
   <packets_received>535636</packets_received>
   <packets_lost>0</packets_lost>
   <frames_received>1088</frames_received>
   <frame_error>0</frame_error>
   <furs_sent>1</furs_sent>
  </stream>
  <stream direction="rx" type="video" context="extended" position="0" encryption_status="
encrypted">
   <codecs>
    <codec>
     <name>H.263+</name>
     <active_time>4</active_time>
     <encrypted_time>4</encrypted_time>
    </codec>
    <codec>
     <name>H.264</name>
     <active_time>1372</active_time>
     <encrypted_time>1372</encrypted_time>
    </codec>
   </codecs>
   <width>768</width>
   <height>448</height>
   <max_bandwidth>2178</max_bandwidth>
   <bandwidth>861</bandwidth>
   <packets_received>27637</packets_received>
   <packets_lost>0</packets_lost>
   <frames_received>13318</frames_received>
   <frame_errors>21</frame_errors>
   <furs_sent>270</furs_sent>
```

```
  </stream>
  <stream direction="rx" type="audio" context="main" position="0" encryption_status="encr
ypted">
   <codecs>
    <codec>
     <name>AAC-LD</name>
     <active_time>1376</active_time>
     <encrypted_time>1376</encrypted_time>
    </codec>
   </codecs>
   <max_bandwidth>64</max_bandwidth>
   <bandwidth>64</bandwidth>
   <packets_received>68751</packets_received>
   <packets_lost>0</packets_lost>
  </stream>
  <stream direction="tx" type="video" context="main" position="0" encryption_status="encr
ypted">
   <codecs>
    <codec>
     <name>H.264</name>
     <active_time>1376</active_time>
     <encrypted_time>1376</encrypted_time>
    </codec>
   </codecs>
   <width>1280</width>
   <height>720</height>
   <max_bandwidth>4005</max_bandwidth>
   <bandwidth>4005</bandwidth>
   <packets_sent>358207</packets_sent>
   <packets_lost>0</packets_lost>
   <frames_sent>12410</frames_sent>
   <frame_error>0</frame_error>
   <furs_received>0</furs_received>
  </stream>
  <stream direction="tx" type="video" context="extended" position="0" encryption_status="
encrypted">
   <codecs>
    <codec>
     <name>H.264</name>
     <active_time>624</active_time>
     <encrypted_time>624</encrypted_time>
    </codec>
   </codecs>
   <width>1280</width>
   <height>720</height>
   <max_bandwidth>1933</max_bandwidth>
   <bandwidth>59</bandwidth>
   <packets_sent>25869</packets_sent>
   <packets_lost>0</packets_lost>
   <frames_sent>12410</frames_sent>
   <frame_error>0</frame_error>
   <furs_received>0</furs_received>
  </stream>
  <stream direction="tx" type="audio" context="main" position="0" encryption_status="encr
ypted">
   <codecs>
    <codec>
     <name>AAC-LD</name>
```

```
     <active_time>1376</active_time>
     <encrypted_time>1376</encrypted_time>
    </codec>
   </codecs>
   <max_bandwidth>64</max_bandwidth>
   <bandwidth>64</bandwidth>
   <packets_sent>68750</packets_sent>
   <packets_lost>0</packets_lost>
  </stream>
 </streams>
</participant_media_summary>
```

Event reference

All the recorded information is nested inside the **`participant_media_summary`** node. The contents of this node are:

Table 24: participant_media_summary XML reference

Nodes and nesting	Attributes	Values / description
participant_guid		Globally Unique Identifier (GUID) of the participant for whom this is the media summary.
call_id		Unique to each constituent call of a participant.
streams		All streams between the TelePresence Server and the participant above are contained within the **`streams`** node.
stream		One **`stream`** node for each stream between TelePresence Server and the participant.
	direction	**`rx`**: the stream is being received by the TelePresence Server. **`tx`**: the stream is being transmitted from the TelePresence Server.
	type	**`audio`**: the stream is formed of packets of encoded audio data. **`video`**: the stream is formed of packets of encoded video data.

Table 24: participant_media_summary XML reference (continued)

Nodes and nesting	Attributes	Values / description
	context	**main**: The default context for a stream if there is only one of that type. **extended**: Used to identify additional streams of the same type. For example, the content channel provides an additional video stream; if an endpoint is sending video and content, then usually the participant's camera provides the **main** stream and the content video is an **extended** stream. Similarly, the audio part of a content presentation is an **extended** stream. On some endpoints you can add in an auxiliary voice participant, which is another example of an **extended audio** stream.
	position	An index which identifies the position of the stream relative to other streams, if necessary. For example, each of the audio streams coming from a three microphone endpoint will have a different index. In the case of three stream audio, **0** is the index given to the leftmost stream, **1** to the center and **2** to the rightmost stream.
	encryption_ status	The encryption status will display as one of the following: **encrypted**, **unencrypted**, **mixed** (when the stream has been both encrypted and unencrypted at some point during the call) or **unknown** (only occurs if a stream has a duration of 0 seconds and it is impossible to determine whether it was encrypted or not).
codecs		Contains a record of all the codecs used on this stream from when it started until the time when this record was taken.
codec		The **codecs** node contains one or more **codec** nodes.

Table 24: participant_media_summary XML reference (continued)

Nodes and nesting			Attributes	Values / description
		name		The name of the codec, e.g. **`H.264`** or **`AAC-LD`**. The full list of possible codecs is: Audio: **`none`**, **`G.711mu`**, **`G.711a`**, **`G.722`**, **`G.728`**, **`G.729`**, **`G.729A`**, **`G.729B`**, **`G.729AB`**, **`G.722.1`**, **`G.723.1`**, **`Polycom (R) Siren14(TM)`**, **`G.722.1C`**, **`AAC-LC`**, **`AAC-LD`**. Video: **`none`**, **`H.261`**, **`H.264`**, **`H.263+`**, **`H.263`**.
		active_time		The number of seconds for which the stream was encoded with this codec.
		encrypted_ time		The number of seconds for which this codec was encrypted.
	width			Width, in pixels, of the frames encoded in the video stream at the time of the record. Not relevant for audio streams.
	height			Height, in pixels, of the frames encoded in the video stream at the time of the record. Not relevant for audio streams.
	max_ bandwidth			The peak bandwidth, in kbps, used by this stream since it started.
	bandwidth			The bandwidth used by this stream, in kbps, at the time of this record.
	packets_ sent			The total number of packets sent to the endpoint. Only present for **`tx`** streams.
	packets_ received			The total number of packets received by the TelePresence Server. Only present for **`rx`** streams.
	packets_ lost			Number of packets lost. Present in both **`rx`** and **`tx`** streams.
	frames_ sent			Number of frames sent to the endpoint. Only present for **`tx`** streams.
	frames_ received			Number of frames received from the endpoint. Only present for **`rx`** streams.
	frame_ errors			Number of frame encode/decode errors. Only present for video streams.
	furs_sent			Number of fast update requests sent to the endpoint Only present for **`rx`** video streams.
	furs_ received			Number of fast update requests received from the endpoint. Only present for **`tx`** video streams.

ISDN Gateway event types

The Cisco TelePresence ISDN Gateway generates the following records:

- new_connection [p.37] when a new connection is initiated.
- connection_proceeding [p.38] when a call has been connected, or a downspeeding event occurs, or a channel is added in an aggregation call.
- multiway_call_transfer [p.40] when an H.323 call leg is transferred to a multiway conference.
- connection_finished [p.43] when a connection is closed (the reason is provided in the event detail).

Note: A CDR record is generated as soon as a call enters the alerting state on the Cisco TelePresence ISDN Gateway, even if the IP side never connects. This includes calls that are queuing for the auto attendant, which are kept in the alerting state while in the queue. Such calls can be identified by their associated call records, which have blank ip_address fields and no video_codec or audio_codec information.

Change summary

Table 25: Changes for Cisco TelePresence ISDN Gateway Version 2.2

Event type	Node	Change
new_connection [p.37]	call	Added support for SIP URIs
connection_finished [p.43]	call	Added (signaling) protocol attribute
	h323_endpoint_details	Added support for SIP URIs
	media_from_isdn / media_to_isdn	Added G.722.1 audio codec
multiway_call_transfer [p.40]	media_from_isdn / media_to_isdn	Added G.722.1 audio codec

new_connection

This event is logged when a call starts.

Example XML

```
<event index="15171028" date="6 January 2011" time="01:57:10" type="new_connection">
  <connection unique_id="1896531">
  </connection>
  <call direction="ip to isdn" calling_number="Codian MCU 4220" original_called_number="
0">
  </call>
  <isdn call_type="bonding" max_call_duration="&lt;no time limit&gt;">
  </isdn>
</event>
```

Event reference

Table 26: new_connection XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection. Generated automatically by the gateway. A positive integer.
call		
	direction	Either **ip to isdn** or **isdn to ip**.
	calling_number	For IP to ISDN calls, the IP endpoint identifier. The identifier can be one of: ■ The H.323 alias or E.164 number (for H.323 connections). ■ The SIP URI (for SIP connections). For ISDN to IP calls, the E.164 number of the ISDN endpoint or **ISDN** if the number is unknown.
	original_called_number	The E.164 number that was originally dialed by the calling endpoint, or **<none>** if an IP endpoint calls the gateway by its IP address.
isdn		
	call_type	One of: ■ **bonding** (Video using bonding) ■ **voice** (Telephone) ■ **h221 aggregation** (Video using N x 64 kbps)
	max_call_duration	The maximum allowed call duration (if any) in seconds. Otherwise **<no time limit>**.

connection_proceeding

This event is logged during a call.

Example XML

Example aggregation call (with three aggregated subcalls so far)

```
<event index="82" date="16 January 2013" time="16:07:24" type="connection_proceeding">
  <connection unique_id="12681023">
  </connection>
  <call via_auto_attendant="no" final_called_number="051000">
  </call>
  <isdn_numbers subcall_0="051000" subcall_1="051000" subcall_2="051000">
  </isdn_numbers>
</event>
```

Example bonding call (with five subcalls)

```
<event index="53" date="16 January 2013" time="16:02:07" type="connection_proceeding">
  <connection unique_id="12681024">
  </connection>
  <call via_auto_attendant="no" final_called_number="10.47.213.52">
  </call>
  <isdn_numbers subcall_0="1" subcall_1="1" subcall_2="1" subcall_3="1" subcall_4="1" >
  </isdn_numbers>
</event>
```

Event reference

Table 27: connection_proceeding XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection as provided in the previous new_ connection event. A positive integer.
call		
	via_auto_attendant	Whether or not the connection was set up via the auto attendant. Either **yes** or **no**.
	final_called_number	The final called number in the format **<ip address:port no>** or a number as generated by the dial plan.

Table 27: connection_proceeding XML reference (continued)

Node	Attribute	Description
isdn_numbers		
	subcall_n	The subcall number in the format **subcall_n**, where **n** is a value between 0 and 28. The number of subcall attributes will match the number of B-channels used by the call, minus 1. The contents of the subcall attribute differ for aggregation and bonding calls. For aggregation calls it contains the full number. For bonding calls it contains only the final digits of the number (that is, those numbers that differ from the master call). For aggregation calls, multiple connection_proceeding events may be logged, up to the number of B-channels used by the call. After the first subcall is aggregated the associated event will indicate just '**subcall_0**'; after the second subcall is aggregated the associated event will indicate '**subcall_0**' and '**subcall_1**'; and so on. If only one channel is connected, no **subcall_n** attributes will be present. For bonding calls, a single connection_proceeding event is logged. The event will have up to 29 subcall attributes. If the subcall numbers are the same as the master call number, the subcall attributes will be blank. Examples of event records for an aggregation call and a bonding call are given above.

multiway_call_transfer

This event is logged when an H.323 call leg is transferred to a multiway conference.

Example XML

```
<event index="189" date="22 June 2011" time="12:21:29" type="multiway_call_transfer">
  <connection unique_id="4623030">
  </connection>
  <call duration="5 mins 35 sec" duration_in_minutes="6" disconnect_reason="call transfer
red to multiway des" new_called_number="7700001@multiway.cisco.com">
  </call>
  <bandwidth number_of_b_channels="6" restricted="no" isdn_bandwidth="384 kbit/s" downspe
eded="no">
  </bandwidth>
  <h323_endpoint_details ip_address="10.11.12.13" dn="456" h323_alias="1025">
  </h323_endpoint_details>
  <media_from_isdn video_codec="H.264" audio_codec="G.722">
  </media_from_isdn>
  <media_to_isdn video_codec="H.264" audio_codec="G.722">
  </media_to_isdn>
</event>
```

Event reference

Table 28: multiway_call_transfer XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection as provided in the previous new_connection event. A positive integer.
call		
	duration	How long the transferred connection lasted, in minutes and seconds.
	duration_in_minutes	How long the transferred connection lasted, rounded up to the nearest minute.
	disconnect_reason	String that specifies why the call was transferred. The string can only be: **call_transferred_to_multiway_destination**.

Table 28: multiway_call_transfer XML reference (continued)

Node	Attribute	Description
bandwidth		
	number_of_ b_channels	Integer that specifies the number of b_channels. To calculate the total ISDN bandwidth of a call, multiply the number of b channels by 64 kbps if **`restricted="no"`** or by 56 kbps if **`restricted="yes"`**.
	restricted	The restricted state of the call at the time that this event was produced, as specified in clause 13/H.242. The restricted state can change during the course of a call. However, observation suggests that endpoints are likely to change at most once, as the call is being set up. Either **`yes`** or **`no`**.
	isdn_ bandwidth	A value in kbps. To calculate the ISDN bandwidth of a call, multiply the number of b channels by 64 kbps if **`restricted="no"`** or by 56 kbps if **`restricted="yes"`**.
	downspeeded	Whether or not the connection was downspeeded. Either **`yes`** or **`no`**.
h323_ endpoint_ details		
	ip_address	The IP address of the IP endpoint.
	dn	Either **`<none>`** or the E.164 number of the IP endpoint.
	h323_alias	The configured endpoint name of the IP endpoint.
media_ from_isdn		
	video_codec	The last non-null video codec at the time this event was created (unless the codec was always null in which case this parameter is also null). One of: ▪ Null ▪ H.261 ▪ H.263 ▪ H.264
	audio_codec	The last non-null audio codec at the time this event was created (unless the codec was always null in which case this parameter is also null). One of: ▪ Null ▪ G.711 ▪ G.722 ▪ G.728 ▪ G.722.1 ▪ G.722.1 Annex C

Table 28: multiway_call_transfer XML reference (continued)

Node	Attribute	Description
media_to_ isdn		
	video_codec	The last non-null video codec at the time this event was created (unless the codec was always null in which case this parameter is also null). One of: ■ Null ■ H.261 ■ H.263 ■ H.264
	audio_codec	The last non-null audio codec at the time this event was created (unless the codec was always null in which case this parameter is also null). One of: ■ Null ■ G.711 ■ G.722 ■ G.728 ■ G.722.1 ■ G.722.1 Annex C

connection_finished

This event is logged when a call completes or is terminated.

Example XML

```
<event index="15171024" date="6 January 2011" time="01:57:10" type="connection_finished">
  <connection unique_id="1896491">
  </connection>
  <call duration="52 sec" duration_in_minutes="1" disconnect_reason="participant ended ca
ll" calling_number="Codian MCU 4220" original_called_number="0" final_called_number="2081
00" direction="ip to isdn">
  </call>
  <bandwidth number_of_b_channels="2" restricted="no" isdn_bandwidth="128 kbit/s" downspe
eded="no">
  </bandwidth>
  <h323_endpoint_details ip_address="10.3.134.94" dn="&lt;none&gt;" h323_alias="Codian MC
U 4220">
  </h323_endpoint_details>
  <media_from_isdn video_codec="H.263" audio_codec="G.728">
  </media_from_isdn>
  <media_to_isdn video_codec="H.263" audio_codec="G.722">
  </media_to_isdn>
</event>
```

Event reference

Table 29: connection_finished XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection as provided in the previous new_connection event. A positive integer.

Table 29: connection_finished XML reference (continued)

Node	Attribute	Description
call		
	duration	How long the connection lasted in minutes and seconds.
	duration_in_minutes	How long the connection lasted rounded up to the nearest minute.
	disconnect_reason	A string that specifies the reason why the participant was disconnected. One of: ■ unspecified ■ no answer ■ rejected ■ rejected immediately ■ busy ■ gatekeeper error ■ protocol error ■ destination unreachable ■ participant ended call ■ participant dropped ■ gatekeeper ended call ■ all participants dropped ■ destination out of order ■ incompatible destination ■ auto attendant idle ■ ip encryption required ■ unknown reason

Table 29: connection_finished XML reference (continued)

Node	Attribute	Description
call (cont)	calling_number	For IP to ISDN calls, the IP endpoint identifier. The identifier can be one of: ▪ The H.323 alias or E.164 number (for H.323 connections). ▪ The SIP URI (for SIP connections). For ISDN to IP calls, the E.164 number of the ISDN endpoint or **ISDN** if the number is unknown.
	original_called_number	The E.164 number that was originally dialed by the calling endpoint, or **<none>** if an IP endpoint calls the gateway by its IP address.
	final_called_number	The final called number in the format **<ip address:port no>** or a number as generated by the dial plan.
	direction	Either **ip to isdn** or **isdn to ip**.
	protocol	The signaling protocol used for the call. Either **h323** or **sip**. This attribute is not displayed when viewing CDR data through the ISDN gateway web user interface. Instead the protocol type is indicated by the presence or absence of SIP or H.323 address fields in the record. For example, the web display for a SIP connection will include a 'SIP endpoint' entry with IP address, URI, and SIP alias attributes.
bandwidth		
	number_of_b_channels	An integer that specifies the number of b_channels. To calculate the total ISDN bandwidth of a call, multiply the number of b channels by 64 kbps if **restricted="no"** or by 56 kbps if **restricted="yes"**.
	restricted	The restricted state of the call at the time that this event was produced, as specified in clause 13/H.242. The restricted state can change during the course of a call (although observation suggests that endpoints are likely to change at most once, as the call is being set up). Either **yes** or **no**.
	isdn_bandwidth	A value in kbps. For example, 128 kbps. To calculate the ISDN bandwidth of a call, multiply the number of b channels by 64 kbps if **restricted="no"** or by 56 kbps if **restricted="yes"**.
	downspeeded	Whether or not the connection was downspeeded. Either **yes** or **no**.

Table 29: connection_finished XML reference (continued)

Node	Attribute	Description
h323_endpoint_details		**Note:** Although this node is named H.323_endpoint_ details, it can actually hold H.323 or SIP data.
	ip_address	The IP address of the endpoint.
	dn	Either **<none>** or the IP endpoint identifier. The identifier can be one of: ■ The E.164 number of the endpoint (H.323 connections). ■ The SIP URI of the endpoint (SIP connections).
	h323_alias	Any alias configured for the IP endpoint. The alias can be one of: ■ A configured name for the endpoint (H.323 connections). ■ A SIP alias for the endpoint (SIP connections).
media_from_isdn		
	video_codec	The last non-null video codec at the time that this event was created (unless the codec was always null in which case this parameter is also null). One of: ■ Null ■ H.261 ■ H.263 ■ H.264
	audio_codec	The last non-null audio codec at the time this event was created (unless the codec was always null in which case this parameter is also null). One of: ■ Null ■ G.711 ■ G.722 ■ G.728 ■ G.722.1 ■ G.722.1 Annex C

Table 29: connection_finished XML reference (continued)

Node	Attribute	Description
media_to_isdn		
	video_codec	The last non-null video codec at the time this event was created (unless the codec was always null in which case this parameter is also null). One of: ▪ Null ▪ H.261 ▪ H.263 ▪ H.264
	audio_codec	The last non-null audio codec at the time this event was created (unless the codec was always null in which case this parameter is also null). One of: ▪ Null ▪ G.711 ▪ G.722 ▪ G.728 ▪ G.722.1 ▪ G.722.1 Annex C

IP Gateway event types

The Cisco TelePresence IP Gateway generates the following records:

- incoming_connection [p.49] when an endpoint has dialed in to the IP gateway and a connection is initiated.
- call_operator [p.50] when an endpoint has dialed in to the IP gateway and the call is connected to the operator.
- outgoing_connection [p.51] when the IP gateway connects to the far end.
- call_rejected [p.52] when the IP gateway's call to the far end is rejected. The far end may be busy, may not answer or rejects the call from the operator.
- call_accepted [p.53] when the far end accepts the call either directly or via the operator.
- enter_menu [p.54] when the caller enters the menu system for the first time.
- video_start [p.55] when the IP gateway plays a video (not a video prompt) to the endpoint.
- video_end [p.56] when the playback of a recording is terminated.
- connection_finished [p.57] when a call completes or is terminated.

incoming_connection

This event is logged when a call starts.

Example XML

```
<event index="102403" date="17 September 2010" time="11:23:17" type="incoming_connectio
n">
  <connection unique_id="27004" calling_number="" original_called_number="">
  </connection>
</event>
```

Event reference

Table 30: incoming_connection XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection in the format **nnnnnnnn**. Generated automatically by the gateway.
	calling_number	The H.323 alias or E.164 number of the endpoint.
	original_called_number	The E.164 number originally dialed by the calling endpoint, or **<none>** if an endpoint calls the IP gateway by its IP address.

call_operator

This event is logged when an operator has been called.

Example XML

```
<event index="102407" date="17 September 2010" time="11:24:25" type="call_operator">
  <connection unique_id="27004">
  </connection>
  <operator user_name="Test_operator">
  </operator>
</event>
```

Event reference

Table 31: call_operator XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection in the format **nnnnnnnn**.
operator		
	user_name	User name of the operator being called.

outgoing_connection

This event is logged when the Cisco TelePresence IP Gateway is connecting through to the far end.

Example XML

```
<event index="102410" date="17 September 2010" time="11:26:03" type="outgoing_connectio
n">
  <connection unique_id="27004">
  </connection>
  <target called_number="10.2.161.253" protocol="h323" gateway="" gatekeeper="">
  </target>
  <screening screened="yes">
  </screening>
</event>
```

Event reference

Table 32: outgoing_connection XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection in the format **nnnnnnnn**.
target		
	called_ number	The number, URI, or IP address of the endpoint being called.
	protocol	The protocol used on the connection. Either **sip** or **h323**.
	gateway	Gateway address. Only present if the protocol is **h323**.
	gatekeeper	Name of the gatekeepers used to make the call. Only present if the protocol is H.323.
	registrar	Name of the registrar used to make the call. Only present if the protocol is SIP.
screening		
	screened	Either **yes** or **no**, depending on whether this outgoing connection is made through the operator.

call_rejected

This event is logged when the far end did not accept the call—by not answering, being busy, or by not accepting the call from the operator.

Example XML

```
<event index="102411" date="17 September 2010" time="11:26:13" type="call_rejected">
  <connection unique_id="27004">
  </connection>
  <target disconnect_reason="participant ended call">
  </target>
</event>
```

Event reference

Table 33: call_rejected XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection in the format nnnnnnnn.
target		
	disconnect_reason	A string explaining the reason why the participant was disconnected. Currently one of: ■ unspecified ■ no answer ■ rejected ■ rejected immediately ■ busy ■ gatekeeper error ■ protocol error ■ destination unreachable ■ participant ended call ■ participant dropped ■ gatekeeper ended call ■ all participants dropped ■ destination out of order ■ incompatible destination ■ auto attendant idle ■ ip encryption required ■ unknown reason

call_accepted

This event is logged when the far end accepted the call (directly or through the operator). After this event, the caller is talking to the far end.

Example XML

```
<event index="102409" date="17 September 2010" time="11:25:40" type="call_accepted">
  <connection unique_id="27004">
  </connection>
  <target name="my_endpoint">
  </target>
</event>
```

Event reference

Table 34: call_accepted XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection in the format **nnnnnnnn**.
target		
	name	Holds the name of the endpoint which accepted the call. If no name is available, it holds the IP address or E.164 number of the endpoint.

enter_menu

This event is logged when a participant is sent into the menu system. This event is seen only when first entering the menu system, not when loading additional menus.

Example XML

```
<event index="102404" date="17 September 2010" time="11:23:17" type="enter_menu">
  <connection unique_id="27004">
  </connection>
  <menu name="Port A menu">
  </menu>
</event>
```

Event reference

Table 35: enter_menu XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection in the format **nnnnnnnn**.
menu		
	name	The name of the menu that was loaded.

video_start

This event is logged when the Cisco TelePresence IP Gateway is playing a video (not a video prompt) to the endpoint.

Example XML

```
<event index="102405" date="17 September 2010" time="11:23:29" type="video_start">
  <connection unique_id="27005">
  </connection>
  <video vcr="videovcr" numeric_id="3505">
  </video>
</event>
```

Event reference

Table 36: video_start XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection in the format nnnnnnnn.
video		
	vcr	The name of the VCR connected to.
	numeric_id	The numeric ID of the recording.

video_end

This event is logged when playback of a recording has terminated.

Example XML

```
<event index="102406" date="17 September 2010" time="11:23:55" type="video_end">
  <connection unique_id="27005">
  </connection>
  <video complete="yes">
  </video>
</event>
```

Event reference

Table 37: video_end XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection in the format **nnnnnnnn**.
video		
	complete	Whether or not the video playback was run to the end. ■ **Yes** - the video ran to completion ■ **No** - the participant terminated the video

connection_finished

This event is logged when a call completes or is terminated.

Example XML

```
<event index="102412" date="17 September 2010" time="11:26:19" type="connection_finished">
  <connection unique_id="27004" duration="2 mins 46 sec" duration_in_minutes="3" disconnect_reason="participant ended call" disconnector="caller">
  </connection>
  <call duration="2 mins 29 sec" duration_in_minutes="3">
  </call>
</event>
```

Event reference

Table 38: connection_finished XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection in the format **nnnnnnnn**.
	duration	How long the connection lasted in minutes and seconds.
	duration_in_minutes	How long the connection lasted rounded up to the nearest minute.
	disconnect_reason	A string explaining the reason why the participant was disconnected. One of: ■ unspecified ■ no answer ■ rejected ■ rejected immediately ■ busy ■ gatekeeper ■ error ■ protocol error ■ destination unreachable ■ participant ended call ■ participant dropped ■ gatekeeper ended call ■ all participants dropped ■ destination out of order ■ incompatible destination ■ auto attendant idle ■ ip encryption required ■ unknown reason
	disconnector	The party that caused the disconnection. One of: ■ caller ■ callee ■ ipgw
call		
	duration	How long the caller was connected to the callee (called party).
	duration_in_minutes	How long the connection lasted rounded up to the nearest minute.

Advanced Media Gateway event types

The Cisco TelePresence Advanced Media Gateway generates the following records:

- connection_started [p.60] when a new connection is initiated.
- connection_finished [p.61] when a connection is terminated.
- participant_disconnected [p.62] when a participant is disconnected (there will be two participant disconnected events per call).

connection_started

This event is logged when a connection is initiated.

Example XML

```
<event index="219151" date="26 April 2010" time="10:38:58" type="connection_started">
  <connection unique_id="860001" />
  <call source="fred" destination="support@cisco.com" />
</event>
```

Event reference

Table 39: connection_started XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection.
call		
	source	Name of the calling participant.
	destination	Name of the called participant.

connection_finished

This event is logged when a connection is terminated.

Example XML

```
<event index="219152" date="26 April 2010" time="10:44:18" type="connection_finished">
  <connection unique_id="860001" />
  <call duration="5 mins 11 sec" duration_in_minutes="6" disconnect_reason="participant e
nded call" />
</event>
```

Event reference

Table 40: connection_finished XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection.
call		
	duration	Duration in minutes and seconds.
	duration_in_minutes	Duration rounded up to nearest minute.
	disconnect_reason	Reason the participant was disconnected. One of: ▪ unspecified ▪ unspecified error ▪ participant ended call ▪ gateway ended call ▪ no answer ▪ rejected ▪ rejected immediately ▪ busy ▪ timeout ▪ network error ▪ protocol error ▪ destination unreachable ▪ authentication failed ▪ service unavailable ▪ capability negotiation error

participant_disconnected

This event is logged when a participant disconnects from, or is disconnected by, the gateway. There are two **participant_disconnected** events per call record.

Example XML


```
<event index="219154" date="26 April 2010" time="10:44:18" type="participant_disconnecte
d">
  <connection unique_id="860001" />
  <endpoint_details ip_address="10.3.129.102" dn="bill" h323_alias="&lt;none&gt;" configu
red_name="&lt;none&gt;" />
  <media_from_endpoint resolution="640 x 480" video_codec="RTVC1" audio_codec="Polycom(R)
Siren7(TM)" bandwidth="2016000 bit/s" />
  <media_to_endpoint resolution="1280 x 720" video_codec="RTVC1" audio_codec="G.722.1" ba
ndwidth="1524000 bit/s" />
</event>
```


Event reference

Table 41: participant_disconnected XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection.
endpoint_details		
	ip_address	The IP address of the proxy server.
	dn	Participant name.

Table 41: participant_disconnected XML reference (continued)

Node	Attribute	Description
`media_from_endpoint` and **`media_to_endpoint`**		
	resolution	The highest resolution received during the call.
	video_codec	The codec used on the outgoing and incoming video. One of: ■ Null ■ H.261 ■ H.263 ■ H.264 ■ RTVC1
	audio_codec	The codec used on the outgoing and incoming audio. One of: ■ Null ■ G.711a ■ G.711mu ■ G.722 ■ G.722.1 ■ G.722.1 Annex C ■ G.723.1 ■ G.728 ■ G.729 ■ G.729A ■ G.729B ■ G.729AB ■ Polycom(R) Siren7(TM) ■ Polycom(R) Siren14(TM)

Serial Gateway event types

The Cisco TelePresence Serial Gateway generates the following records:

- serial_gw_new_connection [p.65] when a new connection is initiated.
- serial_gw_connection_proceeding [p.66] when a call has been connected.
- serial_gw_multiway_call_transfer [p.67] when an H.323 call leg is transferred to a multiway conference.
- serial_gw_connection_finished [p.69] when a connection is terminated.

serial_gw_new_connection

This event is logged when a new connection is initiated.

Example XML

```
<event index="5382255" date="7 April 2011" time="21:17:04" type="serial_gw_new_connectio
n">
        <connection unique_id="1545244">
        </connection>
        <call direction="ip to serial" calling_number="Codian MSE 8510" original_called_number="
4">
        </call>
        <serial max_call_duration="&lt;no time limit&gt;">
        </serial>
</event>
```

Event reference

Table 42: serial_gw_new_connection XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection. A positive integer.
call		
	direction	**ip to serial** or **serial to ip**.
	calling_number	For IP to serial calls, the H.323 alias or E.164 number of the IP endpoint. For serial to IP calls, the gateway port number on which the call arrived.
	original_called_number	For IP to serial calls, the E.164 number that was originally dialed by the calling endpoint or **<none>** if the IP endpoint calls the serial gateway by its IP address. For serial to IP calls there is no original called number and this attribute will contain the gateway port number on which the call arrived.
serial		
	max_call_duration	The maximum allowed call duration (if any) in seconds. Otherwise **<no time limit>**.

serial_gw_connection_proceeding

This event is logged during a call.

Example XML

```
<event index="5382240" date="7 April 2011" time="21:16:54" type="serial_gw_connection_pro
ceeding">
        <connection unique_id="1545238">
        </connection>
        <call via_auto_attendant="no" final_called_number="10.3.135.65!1">
        </call>
</event>
```

Event reference

Table 43: serial_gw_connection_proceeding XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection as provided in the serial_gw_new_connection event. A positive integer.
call		
	via_auto_ attendant	Whether or not the connection was set up via the auto attendant. Either **yes** or **no**.
	final_called_ number	The final called number as generated by the dial plan. In the case of an IP to serial call made without RS-366 dialing, the final called number will be **<none>**.

serial_gw_multiway_call_transfer

This event is logged when an H.323 call leg is transferred to a multiway conference.

Example XML

```
<event index="5382257" date="22 June 2011" time="12:21:29" type="multiway_call_transfer">
  <connection unique_id="4623030">
  </connection>
  <call duration="5 mins 35 sec" duration_in_minutes="6" disconnect_reason="call transfer
red to multiway des" new_called_number="7700001@multiway.cisco.com">
  </call>
  <bandwidth serial_bandwidth="384 kbit/s">
  </bandwidth>
  <h323_endpoint_details ip_address="10.11.12.13" dn="456" h323_alias="1025">
  </h323_endpoint_details>
  <media_from_serial video_codec="H.264" audio_codec="G.722">
  </media_from_serial>
  <media_to_serial video_codec="H.264" audio_codec="G.722">
  </media_to_serial>
</event>
```

Event reference

Table 44: serial_gw_multiway_call_transfer XML reference

Node	Attribute	Description
connection		
	unique_id	Unique identifier for the connection as provided in the serial_gw_new_ connection event. A positive integer.
call		
	duration	How long the transferred connection lasted, in minutes and seconds.
	duration_ in_minutes	How long the transferred connection lasted, rounded up to the nearest minute.
	disconnect_ reason	A string value that specifies the reason why the call was transferred. Can only be: **`call transferred to multiway des`**.
bandwidth		
	serial_ bandwidth	A value in kbps.
h323_endpoint_ details		
	ip_address	The IP address of the endpoint.
	dn	The E.164 number of the IP endpoint or **`<none>`**.
	h323_alias	The configured name of the IP endpoint.

Table 44: serial_gw_multiway_call_transfer XML reference (continued)

Node	Attribute	Description
media_from_serial or media_to_serial		
	video_ codec	The last non-null video codec at the time that this event was created (unless the codec was always null in which case this parameter is also null). One of: ■ Null ■ H.261 ■ H.263 ■ H.264
	audio_ codec	The last non-null audio codec at the time that this event was created (unless the codec was always null in which case this parameter is also null). One of: ■ Null ■ G.711 ■ G.722 ■ G.728 ■ G.722.1 Annex C

serial_gw_connection_finished

This event is logged when a call completes or is terminated.

Example XML

```
<event index="5382182" date="7 April 2011" time="21:16:22" type="serial_gw_connection_fin
ished">
        <connection unique_id="1545209">
        </connection>
        <call duration="16 sec" duration_in_minutes="1" disconnect_reason="unspecified" calling_
number="&lt;none&gt;" original_called_number="03" final_called_number="10.3.135.65!1" dir
ection="serial to ip">
        </call>
        <bandwidth serial_bandwidth="512 kbit/s">
        </bandwidth>
        <h323_endpoint_details ip_address="10.3.135.65" dn="1" h323_alias="Codian MSE 8510">
        </h323_endpoint_details>
        <media_from_serial video_codec="H.264" audio_codec="G.722">
        </media_from_serial>
        <media_to_serial video_codec="H.264" audio_codec="G.722">
        </media_to_serial>
</event>
```

Event reference

Table 45: serial_gw_connection_finished XML reference

Node	Attribute	Description
call		
	duration	How long the connection lasted, in minutes and seconds.
	duration_ in_minutes	How long the connection lasted, rounded up to the nearest minute.
	disconnect_ reason	A string that specifies why the participant was disconnected. One of: ▪ unspecified ▪ unspecified error ▪ participant ended call ▪ gateway ended call ▪ no answer ▪ rejected ▪ rejected immediately ▪ busy ▪ timeout ▪ network error ▪ protocol error ▪ destination unreachable ▪ authentication failed ▪ service unavailable ▪ capability negotiation error

Table 45: serial_gw_connection_finished XML reference (continued)

Node	Attribute	Description
	calling_ number	For IP to serial calls, the H.323 alias or E.164 number of the IP endpoint. For serial to IP calls, the serial gateway port number on which the call arrived.
	original_ called_ number	For IP to serial calls, the E.164 number that was originally dialed by the calling endpoint or **<none>** if the IP endpoint calls the serial gateway by its IP address. For serial to IP calls there is no original called number and this attribute will contain the serial gateway port number on which the call arrived.
	final_ called_ number	The final called number as generated by the dial plan. In the case of an IP to serial call made without RS-366 dialing, the final called number will be **<none>**.
	direction	The direction of the call. Either **`ip to serial`** or **`serial to ip`**.
connection		
	unique_id	Unique identifier for the connection as provided in the serial_gw_new_ connection event. A positive integer.
bandwidth		
	serial_ bandwidth	A value in kbps.
h323_endpoint_ details		
	ip_address	The IP address of the endpoint.
	dn	The E.164 number of the IP endpoint or **<none>**.
	h323_alias	The configured name of the IP endpoint.
media_from_ serial or media_ to_serial		
	video_ codec	The last non-null video codec at the time that this event was created (unless the codec was always null in which case this parameter is also null). One of: ▪ Null ▪ H.261 ▪ H.263 ▪ H.264
	audio_ codec	The last non-null audio codec at the time that this event was created (unless the codec was always null in which case this parameter is also null). One of: ▪ Null ▪ G.711 ▪ G.722 ▪ G.728 ▪ G.722.1 Annex C

Related information

All documentation for the latest versions of the Cisco TelePresence products covered in this guide can be found on Cisco.com.