CHAPTER 10

Understanding Cisco Call Detail Records

This chapter describes the format and logic of the call detail records (CDRs) and call management records (CMRs) that the Cisco Unified Communications Manager Release 6.0(1) (and later) system generates. You can use this information for post-processing activities such as generating billing records and network analysis. The chapter describes how to access the CDR/CMR files and how to interpret fields in the files.

When you install your system, CDRs and CMRs remain disabled by default. You can enable or disable CDRs or CMRs at any time that the system is in operation. You do not need to restart Cisco Unified Communications Manager for the change to take effect. The system responds to all changes within a few seconds. The system enables CMR or diagnostic data separately from CDR data.

This chapter contains the following topics:

CDR Processing

Cisco Unified Communications Manager generates two different types of call information records: CDRs and CMRs. The CDR records store information about a call. The CMR records store information about the quality of the streamed audio of the call. The CDR records relate to the CMR records by way of two GlobalCallID columns: Global CallID callManagerId and GlobalCallID Called. Depending upon the call scenario, there may be more than one CMR for each CDR.

When Cisco Unified Communications Manager places or receives a call, the system generates a CDR record when the call terminates. The system writes the CDR to a flat file (text file). Inside the Cisco Unified Communications Manager, the Call Control process generates CDR records. The system writes records when significant changes occur to a given call, such as ending the call, transferring the call, redirecting the call, splitting the call, joining a call, and so forth.

When CDR records are enabled, Call Control generates one or more CDR records for each call. The system sends these records to EnvProcessCdr, where they are written to the flat files. The number of records that are written varies by type of call, and the call scenario. When Diagnostics are enabled, the device generates CMR records for each call. The system writes one CMR record for each IP phone that is involved in the call, or for each Media Gateway Control Protocol (MGCP) gateway. These records also get sent to EnvProcessCdr where they get written to flat files.

The Cisco Unified Communications Manager generates CDR and CMR records but does not perform any post-processing on the records. The system writes the records to comma-delimited flat files and periodically passes them to the CDR Repository. The CDR and CMR files represent a specific filename format within the flat file.

Filename Format

The following example shows the full format of the filename:

tag_clusterId_nodeId_datetime_seqNumber

- tag—Identifies the type of file, either CDR or CMR
- clusterId—Identifies the server where the Cisco Unified Communications Manager database exists
- nodeId—Identifies the node
- datetime—UTC time in yyyymmddhhmm format
- seqnumber—Sequence number

Two examples of filenames follow:

- cdr_Cluster1_01_200404021658_1
- cmr_Cluster1_02_200404061011_6125

Note For Cisco Unified Communications Manager Business Edition installations, the value assigned to the clusterId equals 01.

Flat File Format

The CDR and CMR flat files have the following format:

- Line 1—List of field names comma separated
- Line 2—List of field type comma separated
- Line 3—Data comma separated
- Line 4—Data comma separated

The following shows an example of a flat file:

```
Line1-"cdrRecordType","globalCallID_callManagerId","globalCallID_callId","origLegCallIdent
ifier",...
Line2-INTEGER,INTEGER,INTEGER,INTEGER,...
Line3-1,1,388289,17586046,...
Line4-1,1,388293,17586054,...
```

Note If the value of the CDR Log Calls With Zero Duration Flag parameter is True, the system writes all calls to a flat file. See the "Configuring CDR Service Parameters" section on page 2-2 for additional information about this parameter.

Cisco Unified Communications Manager CDR Overview

The following sections provide a brief description of how CDRs are generated and managed in Cisco Unified Communications Manager.

- CDR Management, page 10-3
- Types of Call Information Records, page 10-5

CDR Management

The CDR Management (CDRM) feature represents a background application that supports the following capabilities:

- Collects and maintains the CDR/CMR files on the server where you configure CAR.
- Allows third-party applications to retrieve CDR/CMR files on demand through a SOAP interface.
- Accepts on-demand requests for searching file names.
- Sends CDR/CMR files to up to three customer billing servers.
- Monitors disk usage of CDR/CMR files on the server where you configure CAR.
- Periodically deletes CDR/CMR files that were successfully delivered. You can configure the amount of storage used to store flat files. The post-processing applications can later retrieve the buffered historical data to re-get any lost, corrupted, or missing data. The CDRM feature, which is not aware of the flat file format, does not manipulate the file contents.

CDRM comprises two default services, the CDR Agentand the CDR Repository Manager default services, and one activate service, CDR onDemand Service.

CDR Agent

As part of the CDRM feature, a resident component on the server within a Cisco Unified Communications Manager installation acts as the CDR Agent. On this server where both Cisco Unified Communications Manager and the CDR Agent are running, Cisco Unified Communications Manager writes the CDRs into CDR flat files (CSV format). A special control character ("_") that is prefixed to the filename by the call processing module and indicates that the file is not available for transfer. If this control character is not present, the system assumes the file to be available for transfer, and the CDR Agent then sFTPs those files to the designated CDR Repository node. Upon successful transfer, the system deletes the local copy of the file.

Reliability gets the highest priority for the CDRM feature. CDRs are very important financial data, so the goal of this feature is to guarantee that no CDR is lost. The Cisco Unified Communications Manager continuously writes CDRs to flat files, closes existing flat files, and opens new ones. The number of records that are written varies by the type of call and significant changes that occur during a call, such as ending the call, transferring the call, redirecting the call, splitting the call, or joining the call.

CDR Repository Manager

To support CDR management, one instance of the CDR Repository Manager runs on the server where you configure CAR. The CDR Repository Manager manages the CDRs files and can send the files to specified customer/third-party billing servers.

The file gets archived in a directory that is dedicated to the date that is indicated by the UTC timestamp that was placed in the file name when the file was created.

If any external billing server is specified in CDRM configuration, a soft link to the file gets created in a directory that is designated to the destination. The file sender component of the CDR Repository Manager detects this soft link and sends the file to the destination with the specified method. If the delivery is successful, the system removes the soft link in the destination directory.

Every Cisco Unified Communications Manager can generate one CDR file and one CMR file every minute for up to 1 hour. You can configure the maximum disk space that is used for storage of CDR files in the CDR Repository through provisioning.

The File Manager component of the CDR Repository Manager runs hourly. When the File Manager runs, it deletes files with dates outside the configured preservation duration. It also checks whether disk usage has exceeded the high water mark. If so, the system deletes the processed CDR files until the low water mark is reached, starting with the oldest files. However, if any CDR file to be deleted was not successfully sent to the specified billing server, the system leaves it in the CDR Repository and a notification or alarm gets raised. The system creates a flag file during the configured maintenance window, which denies access to the CDR files for the CDR onDemand Service. The system removes the flag file after the maintenance window expires.

For detailed procedures for configuring the CDR Repository Manager and customer billing servers, see the "CDR Repository Manager Configuration" section in the *Cisco Unified Serviceability Administration Guide*.

CDR onDemand Service

The CDR onDemand Service, is a SOAP/HTTPS-based service, that runs on the CDR Repository node. It receives SOAP requests for CDR file name lists based on a user-specified time interval (up to a maximum of 1 hour) and returns all lists that fit the time duration that is specified in the request.

The CDR onDemand Service can also handle requests for delivering a specific CDR file to a specified destination through (s)FTP. The system can activate the CDR onDemand service on the CDR Repository node as it has to access the CDR files in the repository. The system prohibits service during the maintenance window. For detailed information on the CDR onDemand Service, see the *Cisco Unified Communications Manager Developers Guide for Release 6.0(1)*.

Types of Call Information Records

Cisco Unified Communications Manager generates two different types of call information records: Call Detail Records (CDRs) and Call Management Records (CMRs). CDRs store information about the endpoints of the call and other call control/routing aspects. CMRs contain diagnostic information about the quality of the streamed audio and/or video of the call. More than one CMR can exist per CDR.

The CDRs relate to the CMRs via the two globalCallID columns:

- globalCallID_callManagerId
- globalCallId_callId

When the Call Diagnostics service parameter is set to True, the system generates up to two CMRs for each call. Each type of call, such as conference calls, call transfers, forwarded calls, and calls through gateways, produce a set of records that get written to ASCII files at the end of the call. Only completed calls and failed calls generate CDRs and CMRs. Cisco Unified Communications Manager does not perform any post processing on CDRs or CMRs.

This section contains the following topics:

- Global Call Identifier, page 10-5
- Number Translations, page 10-6
- Partitions and Numbers, page 10-6
- Timestamps, page 10-7
- Call Termination Cause Codes, page 10-8

Global Call Identifier

The Cisco Unified Communications Manager allocates a global call identifier (GlobalCallID) each time that a Cisco Unified IP Phone is taken off hook or a call is received from a gateway.

The CDR table (Table 10-1) lists CDRs that are written to the CDR at the end of a call in the order that they are written. GlobalCallIDs for active calls do not appear in the CDR table. Other global IDs also may not appear in the CDR table. For example, each call leg in a conference call gets assigned a GlobalCallID that the conference GlobalCallID overwrites. The original GlobalCallID does not appear in the CDR.

Table 10-1 Sample CDR Table

GlobalCallID	Start Time	End Time
1	973795815	973795820
2	973795840	973795845
5	973795860	973795870
4	973795850	973795880

The CDR table does not contain an entry for GlobalCallID 3 because that call was active when this record was taken. The table shows GlobalCallID 5 listed before GlobalCallId 4 because the GlobalCallID 5 call ended before the GlobalCallID 4 call ended.

Number Translations

The Cisco Unified Communications Manager can perform translations on the digits that a user dials. The translated number, not the actual dialed digits, appears in the CDR.

For example, many companies translate “911” calls to “9-911,” so the caller does not need to dial an outside line in an emergency. In these cases, the CDR contains “9911” even though the user dials “911.”

Note Gateways can perform further modifications to the number before the digits are actually output through the gateway. The CDR does not reflect these modifications.

Partitions and Numbers

Within a CDR, a combination of extension number and partition identifies each phone that is referenced, if partitions are defined. When partitions exist, fully identifying a phone requires both values because extension numbers may not be unique.

The Partition field stays empty when a call ingresses through a gateway. When a call egresses through a gateway, the Partition field shows the partition to which the gateway belongs.

If the dial plan allows callers to use the # key for speed dialing, the # key goes into the database when it is used. For example, the Called Party Number field may contain a value such as “902087569174#.”

In this release, the Party Number fields may include SIP URIs instead of the traditional calling/called party number.

CDRs use the Partition/Extension Numbers shown in Table 10-2:

Table 10-2 Partition/Extension Numbers in CDRs

Phone Number	Description
callingPartyNumber	This party placed the call. For transferred calls, the transferred party becomes the calling party.
originalCalledPartyNumber	This number designates the originally called party, after any digit translations have occurred.
finalCalledPartyNumber	For forwarded calls, this number designates the last party to receive the call. For non-forwarded calls, this field shows the original called party.
lastRedirectDn	For forwarded calls, this field designates the last party to redirect the call. For non-forwarded calls, this field shows the last party to redirect (such as transfer and conference) the call.
callingPartyNumberPartition	This number identifies the partition name that is associated with the CallingPartyNumber field. This field uniquely identifies this number because the Cisco Unified Communications Manager supports multiple Cisco Unified IP Phones with the same extension number in different partitions. For calls that ingress through a gateway, this field remains blank.

Table 10-2 Partition/Extension Numbers in CDRs (continued)

Phone Number	Description
originalCalledPartyNumberPartition	This number identifies the partition name that is associated with the OriginalCalledPartyNumber field. This field uniquely identifies this number because the Cisco Unified Communications Manager supports multiple Cisco Unified IP Phones with the same extension number in different partitions. For calls that egress through a gateway, this field specifies the partition name that is associated with the route pattern that pointed to the gateway.
finalCalledPartyNumberPartition	This number identifies the partition name that is associated with the FinalCalledPartyNumber field. This field uniquely identifies this number because the Cisco Unified Communications Manager supports multiple Cisco Unified IP Phones with the same extension number in different partitions. For calls that egress through a gateway, this field specifies the partition name that is associated with the route pattern that pointed to the gateway.
lastRedirectDnPartition	This number identifies the partition name that is associated with the LastRedirectDn field. This field uniquely identifies this number because the Cisco Unified Communications Manager supports multiple Cisco Unified IP Phones with the same extension number in different partitions. For calls that egress through a gateway, this field specifies the partition name that is associated with the route pattern that pointed to the gateway.

Timestamps

Timestamps within a CDR appear in Universal Coordinated Time (UTC). This value remains independent of daylight saving time changes.

Unsigned 32-bit integers represent all time values. This unsigned integer value displays from the database as a single integer. The field specifies a time_t value that is obtained from the Linux operating system.

The CDR includes the UTC timestamps that are shown in Table 10-3:

Table 10-3 UTC Timestamps in CDRs

Field	Description
dateTimeOrigination	For outgoing calls, this field designates the time the device goes off hook. For incoming calls, this field designates the time the SETUP message is received.
dateTimeConnect	This field designates the time the devices connect and speech begins. This field shows a zero if the call never connects.
dateTimeDisconnect	This field designates the time the call disconnects. This field shows a zero if the call never connects.

Call Termination Cause Codes

The CDR includes two call termination cause codes: OrigCause and DestCause. When the originating party releases the call, the OrigCause gets populated. When the terminating party releases the call, or the call is rejected, the DestCause gets populated. When unpopulated, the termination cause code value shows zero.

Table 10-8 on page 10-109 lists the call termination cause code values per ITU specification Q.850. For On Net call legs, the Cisco Unified Communications Manager determines the call termination cause code value. For Off Net call legs, the far-end switch determines the call termination cause code value.

IP Addresses

The system stores IP addresses as unsigned integers. The CDR file displays IP addresses as signed integers. To convert the signed decimal value to an IP address, first convert the value to a hex number, taking into consideration that it is really an unsigned number. The 32-bit hex value represents four bytes in reverse order (Intel standard). To determine the IP address, reverse the order of the bytes and convert each byte to a decimal number. The resulting four bytes represent the four-byte fields of the IP address in dotted decimal notation.

Note The file displays a negative number when the low byte of the IP address has the most significant bit set.

For example, the IP address 192.168.18.188 displays as -1139627840. To convert this IP address, perform the following procedure:

Step 1 Convert the database display (-1139627840) to a hex value.
The hex value equals 0xBC12A8C0.

Step 2 Reverse the order of the hex bytes, as shown below:
CO A8 12 BC

Step 3 Convert the four bytes from hex to decimal, as shown below:
192 168 18 188

Step 4 The IP address displays in the dotted decimal format:
192.168.18.188

When working with CDRs, you may want to read other tables in the CAR database to obtain information about the type of device in each CDR because the correlation between devices in the Device table and the IP address that is listed in the CDR is not straightforward.

Call Types

A successful call between two parties logs one CDR. Each CDR contains all fields, but some fields may not get used. If a field is not used, see the default values in the CDR definitions table. When supplementary services are involved in a call, additional CDRs may get written.

In addition to the CDR, a call may involve one CMR per endpoint. In a successful call between two parties who are each using an IP phone, two CMRs get written: one for the originator and one for the destination of the call.

This section describes the CDRs that are written for different call types in the system.

Successful On Net Calls

A successful call between two Cisco Unified IP Phones generates a single CDR at the end of the call.

Successful On Net Call CDR Examples

The following table contains two examples:

- A—A 60-second call that the caller terminates
- B—A 60-second call that the called party clears

	Calling Party	Calling Partition	Original Called Party	Original Called Partition	Orig Cause	Dest Cause	Duration
A	2001	Accounts	2309	Marketing	16	0	60
B	2001	Accounts	2309	Marketing	0	16	60

Abandoned Calls

The logging of calls with zero duration represents an optional action. If logging calls with zero duration is enabled, the following actions occur:

- All calls generate a CDR.
- If the call is abandoned, such as when a phone is taken off hook and placed back on hook, various fields do not contain data. In this case, the originalCalledPartyNumber, finalCalledPartyNumber, the partitions that are associated with them, the destIpAddr, and the dateTimeConnect fields all remain blank. All calls that are not connected have a duration of 0 second. When a call is abandoned, the cause code contains **0**.
- If the user dials a directory number and abandons the call before it connects, the FirstDest and FinalDest fields and their associated partitions contain the directory number and the partition to which the call would have been extended. The DestIp field remains blank, and the duration specifies 0 second.

Abandoned Calls CDR Examples

The following table contains two examples:

- A—Extension 2001 goes off hook then on hook (when the CdrLogCallsWithZeroDurationFlag is set to **True**).
- B—Extension 2001 calls 2309, but 2001 hangs up (abandons) the call before it is answered.

	Calling Party	Calling Partition	Original Called Party	Original Called Partition	Orig Cause	Dest Cause	Duration
A	2001	Accounts			16	0	0
B	2001	Accounts	2309		16	0	0

Calls with Busy or Bad Destinations

The system logs all these calls as normal calls with all relevant fields containing data. The Calling or Called Party Cause fields contain a cause code that indicates why the call was not connected, and the Called Party IP and Date/Time Connect fields remain blank. The system logs all unsuccessful calls, even if zero duration calls are not being logged (CdrLogCallsWithZeroDurationFlag set at **True** or **False**, a duration of zero, and a DateTimeConnect value of zero).

Calls with Busy or Bad Destinations CDR Examples

The following table contains three examples:

- A—Call to PSTN number; party is engaged (cause 17 = user busy).
- B—Call to PSTN number; number does not exist (cause 1 = number unavailable).
- C—Call to PSTN fails because PSTN trunks are out of order (cause 38 = Network Out Of Order).

	Calling Party	Calling Partition	Original Called Party	Original Called Partition	Orig Cause	Dest Cause	Duration
A	2001	Accounts	902920262226	PSTN	0	17	0
B	2001	Accounts	902920100000	PSTN	0	1	0
C	2001	Accounts	902920262226	PSTN	0	38	0

Short Calls

A short call, with a CdrLogCallsWithZeroDurationFlag set at True and a duration of less than 1 second, appears as a zero duration call in the CDR. The DateTimeConnect field, which shows the actual connect time of the call, differentiates these calls from failed calls. For failed calls (which never connected), this value equals zero.

Short Call CDR Example

The following table contains an example of a successful On Net call with a duration of less than 1 second, that the called party cleared.

Calling Party	Calling Partition	Original Called Party	Original Called Partition	Orig Cause	Dest Cause	DateTime Connect	Duration
2001	Accounts	2309	Marketing	0	16	973795815	0

Forwarded or Redirected Calls

Forwarded calls generate a single CDR and show the Calling Party, Original Called Number, Last Redirecting Number, Final Called Number, and the associated partitions. If the call is forwarded more than twice, the intermediate forwarding parties do not populate in the CDR.

Call forwarding can occur on several conditions (always, on busy, and on no answer). The condition under which the call is forwarded does not populate in the CDR.

The CDRs for forwarded calls match those for normal calls, except for the originalCalledPartyNumber field and the originalCalledPartyNumberPartition field. These fields contain the directory number and partition for the destination that was originally dialed by the originator of the call. If the call gets forwarded, the finalCalledPartyNumber and finalCalledPartyNumberPartition fields differ and contain the directory number and partition of the final destination of the call.

Also, when a call is forwarded, the lastRedirectDn and lastRedirectDnPartition fields contain the directory number and partition of the last phone that forwarded or redirected the call.

Forward or Redirected Call CDR Examples

The following table contains two examples:

- A—Call from the PSTN to extension 2001, forwarded to 2309, where the call is answered
- B—Call from the PSTN to extension 2001, forwarded to 2309, which forwards to voice-messaging system

	Calling Party	Original Called Party	Original Called Partition	Final Called Party	Final Called Partition	Last Redirect Party	Last Redirect Partition	Duration	OriginalCalled Party Redirect OnBehalfOf	Last Redirect Redirect OnBehalfOf
A	02920262227	2001	ACNTS	2309	MKTG	2001	ACNTS	120	5	5
B	02920262227	2001	ACNTS	6000	VMAIL	2309	MKTG	60	5	5

Pickup Calls

Cisco Unified Communications Manager includes two pickup modes: Pickup and Auto Pickup. The following sections describe these calls:

- Pickup, page 10-12
- Auto Pickup, page 10-12

Pickup

Pickup calls work like forwarded calls. The CDRs for pickup calls match those for normal calls except for the originalCalledPartyNumber field and the originalCalledPartyNumberPartition field. These fields contain the Directory Number and partition for the destination that was originally dialed by the originator of the call.

If the call is picked up, the finalCalledPartyNumber and finalCalledpartyNumberPartition fields will differ and contain the Directory Number and partition of the phone that picked up the call. Also, when a call is picked up, the lastRedirectDn and lastRedirectDnPartition fields contain the directory number and partition of the last phone that redirected this call.

The origTermination, destTermination, lastRedirect, and Join OnBehalfOf fields contain 16 (Pickup) and the redirect reason field contains 5 (Pickup).

Pickup CDRs look the same for all types of pickup: Pickup, Group Pickup and Other Pickup.

Pickup Call CDR Example

1. A call comes in from the PSTN to extensions 2000, 2001, and 2002, which are in the same pickup group.
2. Extension 2002 picks up the call that is ringing on 2001.
3. Extension 2002 answers the call, and the call connects between the PSTN caller and extension 2002.

Call ID	Orig Cause	Calling Party	Dest Cause	Original Called Party	Final Called Party	Last Redirect Party	Orig Termina-tion On BehalfOf	Dest Termina-tion On BehalfOf	Last Redirect On BehalfOf	Last Redirect Reason	Join On BehalfOf
22	0	9728131234	16	2001	2002	2001	16	16	16	5	16

Auto Pickup

Auto Pickup works like call pickup with auto answer. The call connects automatically, so no need exists for the last answer softkey press. The system generates two CDRs for Auto Pickup, and these CDRs have the same Call ID.

The system generates the first CDR for the original call. This CDR will have the origTerminationOnBehalfOf and destTerminationOnBehalfOf fields equal to 16 (Pickup), which indicates that the call terminated on behalf of the pickup feature.

The second CDR represents the final call after it was picked up. This CDR will have the lastRedirectOnBehalfOf and the joinOnBehalfOf fields set to 16 (Pickup), which indicates that the system joined the call on behalf of the Pickup feature. The lastRedirectReason contains the redirect reason of 5 (Pickup).

Auto Pickup CDRs look the same for all types of auto pickup: Auto Pickup, Auto Group Pickup, and Auto Other Pickup.

Auto Pickup CDR Example

1. A call comes in from the PSTN to extension 2001; 2001 and 2002 reside in the same pickup group.
2. Extension 2002 picks up the call that is ringing on 2001.
3. The call automatically connects between the PSTN caller and extension 2002.

Call ID	Orig Cause	Calling Party	Dest Cause	Original Called Party	Final Called Party	Last Redirect Party	Orig Termina-tion On BehalfOf	Dest Termina-tion On BehalfOf	Last Redirect On BehalfOf	Last Redirect Reason	Join On BehalfOf
11	126	9728131234	126	2001	2001	2001	16	16	0	0	0
11	0	9728131234	16	2002	2002	2001	16	16	16	5	16

Transferred Calls

A single CDR cannot show all the data that is necessary for a call transfer because it is too complex. Each time that a call is transferred, the Cisco Unified Communications Manager terminates the CDR for that call and initiates a new CDR.

Calls that are transferred have multiple CDRs logged for them, as follows:

1. Original call from party A to party B.
2. Call from the transferring party (party A or B) to the transfer destination (party C).
3. Call from the transferred party (party A or B) to the destination (party C).

The first CDR represents the original placed call. The second CDR represents the setup call (consultative/announcement) that is used to initiate the transfer. The third CDR represents the transferred call itself. The first two CDRs have the origCause_value and destCause_value set to Split (126).

They also have the origCallTerminationOnBehalfOf and destCallTerminationOnBehalfOf fields set to Transfer (10) to indicate that these calls were involved in a transfer. The transferred leg of the call has the joinOnBehalfOf field set to Transfer (10) to indicate this call resulted from a transfer. Therefore, all legs of the transfer can be tied back to a single call.

Transferred Calls CDR Examples

The following examples which are not an exhaustive set, illustrate the records that would get generated under the stated circumstances. This example should help clarify what records get generated on transferred calls.

Example 1

A calls B, A transfers B to C. The following three calls get logged:

1. Call from A to B
2. Call from A to C
3. Call from B to C

If the call was a blind transfer, the call from A to C will have a duration of zero seconds. If the call was a consultation transfer, all calls will have non-zero durations. Original Called Party and Call Party Number fields register the same.

Example 2

A calls B; B transfers A to C. The following three calls get logged:

1. Call from A to B
2. Call from B to C
3. Call from A to C

If the call was a blind transfer, the call from B to C will have a duration of zero seconds. If the call was a consultation transfer, all calls will have non-zero durations. Original Called Party and Call Party Number fields register the same.

Example 3

A calls B, B transfers A to C on a blind transfer. C gets Call Forwarded on No Answer to D. The following calls get logged:

1. Call from A to B
2. Call from B to C
3. Call from A to D

Because the call was a blind transfer, the call from B to C has a duration of zero seconds. The call from A to D will have the Original Called Party field set to “C”, and the Called Party Number field gets set to “D”.

Transfer Without Consultation

The process of transferring a call, without consultation, involves the creation of three CDRs. The first CDR reflects the call between the original two parties (A and B), the second CDR represents the (zero length) call between the transferring party (A) and the new party (C), and the final CDR reflects the call between B and C.

No CDR reflects the time that a call is on hold. If a call is through a PSTN gateway, the call accrues charges that are not reflected in the CDRs while the call is on hold.

Transfer Without Consultation CDR Examples

The following table contains three examples:

- A—Call from extension 2001 to a PSTN number sustains talking for 120 seconds.
- B—Extension 2001 initiates a transfer without consultation (duration is zero) to extension 2002.
- C—Extension 2001 completes the transfer, dropping out of the call, and leaving a call between the other two parties.

	Calling Party	Calling Partition	Calling Leg	Original Called Party	Original Called Partition	Called Leg	Orig Cause	Dest Cause	OrigCall Term On BehalfOf	DestCall Term On BehalfOf	Join On BehalfOf	Duration
A	2001	ACNTS	101	3071111	PSTN	102	126	126	10	10	0	120
B	2001	ACNTS	103	2002	ACNTS	104	126	126	10	10	0	0
C	3071111	PSTN	102	2002	ACNTS	104	0	16	0	0	10	350

Transfer with Consultation

Transfer with Consultation essentially acts identical to Transfer Without Consultation, except the duration of the middle call is not zero.

As with a Transfer Without Consultation, Cisco Unified Communications Manager creates three CDRs. The first CDR reflects the call between the original two parties (A and B), the second CDR represents the consultation call between the transferring party (A) and the new party (C), and the final CDR reflects the call between B and C.

Transfer with Consultation CDR Examples

The following table contains three examples:

- A—Call from extension 2001 to a PSTN number sustains talking for 120 seconds.
- B—Extension 2001 places the PSTN call on hold and calls extension 2002, talking for 30 seconds.
- C—Extension 2001 completes the transfer, dropping out of the call, leaving a call between the other two parties.

	Calling Party	Calling Partition	Calling Leg	Original Called Party	Original Called Partition	Called Leg	Orig Cause	Dest Cause	OrigCall Term On BehalfOf	DestCall Term On BehalfOf	Join On BehalfOf	Duration
A	2001	ACNTS	101	3071111	PSTN	102	126	126	10	10	0	120
B	2001	ACNTS	103	2002	ACNTS	104	126	126	10	10	0	30
C	3071111	PSTN	102	2002	ACNTS	104	0	16	0	0	10	350

Conference Calls

Three major operational factors exist for conference call CDRs:

1. When the conference decreases to two parties, the two parties connect directly and release the conference resource. This change generates an additional CDR for the call between the last two parties in the conference call.

   For example, if four people are connected in a conference call (Amy, Dustin, Spencer, Ethan), when Ethan hangs up, three people remain in the conference call that is connected to the conference bridge (Amy, Dustin, Spencer). When Spencer hangs up, only two people remain in the conference call (Amy and Dustin). The system joins Amy and Dustin directly, and the conference resource gets released. Directly joining Amy and Dustin creates an additional CDR between the last two parties in the conference.

2. The system adds the conference controller information to the comment field in the CDR. This information identifies the conference controller. No need now exists to examine the consultation call to determine who is the conference controller. The following example shows this information:

   Comment field = “ConfControllerDn=1000;ConfControllerDeviceName=SEP0003E333FEBD”

- The conference controller DN + conference controller device name uniquely identify the conference controller. A need for the device name exists in the case of shared lines.
- If the call is involved in multiple conference calls, the comment field contains multiple conference controller information. This situation may occur when the conference goes down to two parties, and one of these parties starts another conference. If this is the case, the last conference controller information in the comment field identifies the conference controller.

3. The party that added the participant, known as the requestor party, appears in the CDR comment field. The tags for the requestor information include ConfRequestorDn and ConfRequestorDeviceName. The party that requested to remove a participant, known as the drop requestor, appears in the CDR comment field. The tags for the drop requestor information include DropConfRequestorDn and DropConRequestorDeviceName.

Calls that are part of a conference have multiple records that are logged for them. The number of CDRs that are generated depends on the number of parties in the conference. One CDR exists for each party in the conference, one CDR for the original placed call, and one CDR for each setup call that is used to join other parties to the conference. Therefore, for a three-party ad hoc conference, six CDRs exist:

- One CDR for the original call
- Three CDRs for the parties that are connected to the conference
- One CDR for each setup call
- One CDR for the final two parties in the conference

You can associate the setup calls with the correct call leg in the conference by examining the calling leg ID and the called leg ID.

The conference bridge device holds special significance to the Cisco Unified Communications Manager. Calls to the conference bridge appear as calls to the conference bridge device. A special number in the form “b0019901001” shows the conference bridge port. All calls get shown “into” the conference bridge, regardless of the actual direction. You can determine the original direction of each call by examining the setup call CDRs.

The call legs that are connected to the conference have the following values for these fields:

- finalCalledPartyNumber—Represents a conference bridge “b0019901001”
- origCalledPartyRedirectOnBehalfOf—Set to Conference (4)
- lastRedirectRedirectOnBehalfOf—Set to Conference (4)
- joinOnBehalfOf—Set to Conference (4)
- comment—Identifies the conference controller

The original placed call and all setup calls that were used to join parties to the conference have the following values for the fields:

- origCallTerminationOnBehalfOf—Set to Conference (4).
- destCallTerminationOnBehalfOf—Set to Conference (4).

Conference Calls CDR Examples

The following tables contain these examples:

- Call from 2001 to 2309.
- After 60 seconds, user 2001 presses the “conference” key on the Cisco Unified IP Phone and dials the PSTN number “3071111.”
- 3071111 answers and talks for 20 seconds; then, 2001 presses the conference key to complete the conference.
- The conference talks for 360 seconds.
- Each call leg shows as a call into the conference bridge. The call appears as a call ***into*** the bridge, regardless of the actual direction of the call.

- 3071111 hangs up and leaves 2001 and 2309 in the conference. Because only two participants remain in the conference, the conference features directly join the two, and they talk for another 55 seconds.

Calling Party	Calling Partition	Calling Leg	Original Called Party	Original Called Partition	Called Leg	Final Called Party	Final Called Partition	Last Redirect Party	Last Redirect Reason	Orig Conversation Id
2001	ACNTS	101	2309	MKTG	102	2309	MKTG	2001	0	0
2001	ACNTS	101	2309	MKTG	115	b0029901001		b0029901001	0	1
2309	ACNTS	101	b0029901001		116	b0029901001		b0029901001	0	1
3071111	PSTN	101	b0029901001		117	b0029901001		b0029901001	0	1
2001	ACNTS	105	3071111	PSTN	106	3071111	PSTN	3071111	0	0
2001	ACNTS	101	2309	MKTG	102	2309	MKTG	b0029901001	98	0v

OrigCall Termination OnBehalfOf	DestCall Termination OnBehalfOf	Original CalledParty Redirect OnBehalfOf	Last Redirect OnBehalfOf	Join OnBehalfOf	Duration	Comment
4	4	0	0	0	60	
12	0	4	4	4	360	ConfControllerDn=2001;ConfControllerDeviceName=SEP0003E333FEBD
12	0	4	4	4	360	ConfControllerDn=2001;ConfControllerDeviceName=SEP0003E333FEBD
4	4	4	4	4	360	ConfControllerDn=2001;ConfControllerDeviceName=SEP0003E333FEBD
4	4	0	0	0	20	
12	42	0	4	4	55	ConfControllerDn=2001;ConfControllerDeviceName=SEP0003E333FEBD

Meet-Me Conferences

A meet-me conference occurs when several parties individually dial into a conference bridge at a predetermined time.

The Cisco Secure Conference feature uses the existing callSecuredStatus field to display the highest security status that a call reaches. For meet-me conferences, the system clears calls that try to join the conference but do not meet the security level of the meet-me conference with a terminate cause = 58 (Bearer capability not presently available).

Meet-Me Conference CDR Examples

The following table contains an example CDR for the following scenario. 5001 specifies the dial-in number. The conference bridge device signifies special significance to the Cisco Unified Communications Manager, and calls to the conference bridge appear as forwarded calls; that is, User A phones the predetermined number (5001), and the call gets forwarded to a conference bridge port. The conference bridge port appears with a special number of the form "b0019901001."

- User A (2001) calls into a meet-me conference bridge with the phone number 5001.
- User B (2002) calls into a meet-me conference bridge with the phone number 5001.
- User C (2003) calls into a meet-me conference bridge with the phone number 5001.

	Calling Party	Calling Partition	Original Called Party	Original Called Partition	Final Called Party	Final Called Partition	Last Redirect Party	Last Redirect Partition	Duration
A	2001	Accounts	5001		b0019901001		b0019901001		70
B	2002	Accounts	5001		b0019901001		b0019901001		65
C	2003	Accounts	5001		b0019901001		b0019901001		80

Ad Hoc Conference Linking

The advanced ad hoc conference linking feature allows you to link multiple ad hoc conferences together by adding an ad hoc conference to another ad hoc conference as if it were an individual participant. You can also use the methods that are available for adding individual participants to an ad hoc conference to add another conference to an ad hoc conference.

CDRs that the advanced ad hoc conference linking feature generates include a field called OrigConversationId. This field associates the conference bridges that are involved in a linked conference. The Comment field of the CDR adds the ConfRequestorDN and ConfRequestorDeviceName tags to indicate add/drop of participants of the conference by a non-controller of the conference.

Two types of conference linking exist:

- Linear—No more than two ad hoc conferences can link directly to any participating conference.
- Nonlinear—Three or more ad hoc conferences that link directly to another conference. The system does not permit this type of linking by default because potentially negative impact on conference resources exists.

Linear Ad Hoc Conference Linking Using Join CDR Example

The following table contains example CDRs for this scenario:

- Alice (1000) calls Bob (1001). This is an original call.
- Bob (1001) conferences in Carol (1002) This is a consultation call.
- Dave (1003) calls Carol (1002). This is an original call.
- Dave (1003) conferences in Ed (1004) This is a consultation call.
- Two separate conferences are created. Carol is in both conferences. At this point CDR1, CDR2, CDR3, and CDR4 are generated.
- Carol (1002) joins the two conferences through a conference bridge (b002990122). At this point CDR5 is generated.
- Dave (1003) joins the two conferences through a conference bridge (b002990122). At this point CDR6 is generated.
- Ed (1004) leaves the conference. CDR7 is generated.
- Dave (b002990122) leaves the conference. CDR8 is generated.
- Alice (1000) leaves the conference. CDR9 is generated.
- Bob (1001) leaves the conference. CDR10 is generated.

- Carol (1002) leaves the conference. CDR11 is generated.

Calling Party Number	globalCallID-callid	Original Leg Call Identifier	Dest Leg Call Identifier	Original Called Party Number	Final Called Party Number	Last RedirectDn	OrigCall Termination OnBehalfOf
1000 (CDR1)	1	11	12	1001	1001	1001	4
1001 (CDR2)	2	13	14	1002	1002	1002	4
1003 (CDR3)	3	21	22	1002	1002	1002	4
1003 (CDR4)	4	23	24	1004	1004	1004	4
1002 (CDR5)	3	22	25	b0029901222	b0029901222	1003	4
1003 (CDR6)	3	21	26	b0029901222	b0029901222	1003	0
1004 (CDR7)	3	24	27	b0029901222	b0029901222	1003	0
b0029901222 (CDR8)	1	25	28	b0029901001	b0029901001	10020	0
1000 (CDR9)	1	11	15	b0029901001	b0029901001	1001	0
1001 (CDR10)	1	12	16	b0029901001	b0029901001	1001	0
1002 (CDR11)	1	14	17	b0029901001	b0029901001	1001	0

This is a continuation of the previous table.

Calling Party Number	DestCall Termination OnBehalfOf	LastRedirect Redirect Reason	LastRedirect Redirect OnBehalfOf	Original ConversationID	Destination Conversation ID	Comment
1000 (CDR1)	4	0	0	0	0	
1001 (CDR2)	4	0	0	0	0	
1003 (CDR3)	4	0	0	0	0	
1003 (CDR4)	4	0	0	0	0	

Calling Party Number	DestCall Termination OnBehalfOf	LastRedirect Redirect Reason	LastRedirect Redirect OnBehalfOf	Original ConversationID	Destination Conversation ID	Comment
1002 (CDR5)	4	98	4	0	2222	ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1
1003 (CDR6)	0	98	4	0	2222	ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1
1004 (CDR7)	0	98	4	0	2222	ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333 FAD1
B0029901222 (CDR8)	0	98	4	2222	1111	ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333 FAD1
1000 (CDR9)	0	98	4			
1001 (CDR10)	0	98	4			
1002 (CDR11)	0	98	4			

Precedence Calls (MLPP)

Precedence calls take place the same as other calls except the precedence level fields get set in the CDR. Also, when a higher level precedence call preempts a call, the cause codes indicate the reason for the preemption.

Precedence Calls CDR Example

The following table contains an example CDR for this scenario:

- User A (2001) calls another IP phone by dialing a precedence pattern (precedence level 2).
- User A (2001) calls another IP phone by dialing a precedence pattern (precedence level 3).
- User A receives a higher level precedence call from another network (precedence level 1).
- The higher precedence level call preempts the first call.

Calling Party	Calling Partition	Origin Precedence Level	Original Called Party	Original Called Partition	Dest Precedence Level	Orig Cause	Dest Cause
2001	CMD	2	826001	FIRE	2	0	16
2001	CMD	3	836001	FIRE	3	0	16
9728552001	GEN	1	6001	FIRE	1	16	0
2001	CMD	2	826001	FIRE	2	0	9
9728552001	GEN	1	826001	FIRE	1	0	16

Malicious Calls

When a call gets identified as a malicious call (button press), the local Cisco Unified Communications Manager network flags the call. The Comment field flags the malicious call.

Malicious Calls CDR Example

The following table contains an example CDR of a customer call that gets marked as malicious.

Calling Party	Calling Partition	Original Called Party	Original Called Partition	Orig Cause	Dest Cause	Comment
9728552001	CUST	5555	ACNTS	0	16	“callFlag=MALICIOUS”

Conference Drop Any Party

The Conference Drop Any Party feature terminates calls that look the same as other calls except for a new cause code. The cause code identifies calls that get terminated by this feature.

Conference Drop Any Party CDR Example

The following table contains an example CDR for a call that was connected to a conference and dropped by this feature.

Calling Party	Calling Partition	Original Called Party	Orig Cause	Original Called Partition	Called Leg	Dest Cause	Final Called Party	Final Called Partition	Last Redirect Party
2001	ACNTS	2309	0	MKTG	102	16	2309	MKTG	2001
2001	ACNTS	2309	16	MKTG	115	0	b0029901001		b0029901001
2309	ACNTS	b0029901001	0		116	128	b0029901001		b0029901001
3071111	PSTN	b0029901001	16		117	0	b0029901001		b0029901001
2001	ACNTS	2309	16	PSTN	106	0	3071111	PSTN	30711111

Orig Conversation ID	OrigCall Termination OnBehalfOf	DestCall Termination OnBehalfOf	OriginalCalled Pty Redirect OnBehalfOf	LastRedirect Redirect OnBehalfOf	Join OnBehalfOf	Duration
0	4	4	0	0	0	60
1	12	0	4	4	4	360
1	13	0	4	4	4	200
1	4	4	4	4	4	360
0	4	4	0	0	0	20

Immediate Divert (to Voicemail)

CDRs for Immediate Divert calls take place the same as forwarded calls except values exist for **origCalledPartyRedirectOnBehalfOf** and the **lastRedirectRedirectOnBehalfOf** fields.

Immediate Divert CDR Example

The following table contains an example CDR for this scenario:

Calling Party	Calling Partitio n	Original Called Party	Original Called Partition	Final Called Party	Final Called Partition	Last Redirect Party	Last Redirect Partition	Duration	OrigCalled Party Redirected OnBehalfOf	Last Redirect Redirect OnBehalfOf
02920262227		2001	ACNTS	2309	MKTG	2001	ACNTS	120	5	5
02920262227		2001	ACNTS	6000	VMAIL	2309	MKTG	60	5	5

Video Calls

The following table contains an example CDR for a video call for this scenario:

- Calling party 51234 calls the called party 57890.
- 100 = H.261
- 187962284 = 172.19.52.11
- 288625580 = 172.19.52.17
- 320 - 320

Video Calls CDR Example

- 2 = QCIF

Calling Party	Calling Partition	Calling Leg	Original Called Party	Original Called Partition	Called Leg	Orig VideoCap_ Codec	Orig VideoCap_ Bandwidth	Orig VideoCap_ Resolution	OrigVideo Transport Address_IP	OrigVideo Transport Address_Port
51234	CISCO	101	57890	CISCO	102	100	320	2	187962284	49208

Dest VideoCap_ Codec	Dest VideoCap_ Bandwidth	Dest VideoCap_ Resolution	DestVideo Transport Address_IP	DestVideo Transport Address_Port
100	320	2	288625580	49254

Call Monitoring and Call Recording

The system generates CDRs for the Call Monitoring and Call Recording features by using existing CDR fields.

For both monitoring and recording, the monitoring calls and recording calls have one-way media. The media fields stay empty for one side of the call for one-way media CDRs.

The **destConversationID** field of the Call Monitoring CDR matches the agent call leg identifier in the CDR of the call that is monitored and links together the Call Monitoring CDR and the CDR of the monitored call.

The **origConversationID** field of the two Call Recording CDRs matches the agent call leg identifier in the Recording Call CDR and links together the Call Recording CDR and the CDR of the recorded call.

Call Monitoring CDR Examples

The following table contains example CDRs for a monitor call for the following scenarios:

- Example A—Customer 9728134987 calls the agent 30000, and the agent answers. Supervisor 40003 monitors the call. The **destConversationID** from the monitoring call matches the **destLegCallIdentifier** of the monitored call.
- Example B—Agent 30000 calls the customer 9728134987, and the customer answers. The supervisor 40003 monitors the call. The **destConversationID** from the monitoring call matches the **origLegCallIdentifier** of the monitored call.

	Global Call ID callid	Orig Leg Call Identifier	Dest Leg Call Identifier	Calling Party Number	Orig Called Party Number	Final Called Party Number	Last RedirectDn	Orig Cause Values
A—Monitored Call	7	16777230	16777231	9728134987	30000	30000	30000	16
A—Monitoring Call	10	16777232	16777235	4003	b001501001	b001501001	b001501001	0
B—Monitored Call	71	16777299	16777300	30000	9728134987	9728134987	9728134987	16
B—Monitoring Call	101	16777932	16777935	40003	b001501002	b001501002	b001501002	0

Dest Cause Value	Orig Called Party Redirect Reason	last Redirect Redirect Reason	Orig Called Party Redirect OnBehalfOf	last Redirect Redirect OnBehalfOf	dest Conversation ID
0	0	0			0
0	370	370	28	28	16777231

Dest Cause Value	Orig Called Party Redirect Reason	last Redirect Redirect Reason	Orig Called Party Redirect OnBehalfOf	last Redirect Redirect OnBehalfOf	dest Conversation ID
0	0	0			0
0	370	370	28	28	16777299

Call Recording CDR Examples

The following table contains example CDRs for recording calls for the following scenarios:

- Example A—Customer 9728134987 calls the agent 30000, and the agent answers. The recording feature creates two recording calls to the recording device. This action results in two additional CDRs: one for the agent voice and another for the customer voice. The **origConversationID** from the recording CDRs match the **destLegCallIdentifier** of the recorded CDR. In this example, the customer hangs up.
- Example B—Agent 30000 calls the customer 9728134987, and the customer answers. The recording feature creates two recording calls to the recording device. This action results in two additional CDRs: one for the agent voice and another for the customer voice. The **origConversationID** from the recording CDRs matches the **origLegCallIdentifier** of the recorded CDR. In this example, the agent hangs up.

	Global Call ID callid	Orig Leg Call Identifier	Dest Leg Call Identifier	Calling Party Number	Orig Called Party Number	Final Called Party Number	Last RedirectDn	Orig Cause Values
A—Recorded Call	7	16777110	16777111	9728134987	30000	30000	30000	16
A—Recording Call CDR1	10	16777120	16777121	30000	90000	90000	90000	0
A—Recording Call CDR2	11	16777122	16777123	30000	90000	90000	90000	0
B—Recorded Call	71	16777113	16777114	30000	9728134987	9728134987	9728134987	16
B—Recording Call CDR1	100	16777220	16777221	30000	90000	90000	90000	16
B—Recording Call CDR2	110	16777222	16777223	30000	90000	90000	90000	16

Dest Cause Value	Orig Called Party Redirect Reason	last Redirect Redirect Reason	Orig Called Party Redirect OnBehalfOf	last Redirect Redirect OnBehalfOf	Orig Conversation ID
0	0	0			0
0	354	354	27	27	16777111
0	354	354	27	27	16777111
0	0	0			0

Dest Cause Value	Orig Called Party Redirect Reason	last Redirect Redirect Reason	Orig Called Party Redirect OnBehalfOf	last Redirect Redirect OnBehalfOf	Orig Conversation ID
0	354	354	27	27	16777113
0	354	354	27	27	16777113

AAC and iLBC Calls

The Advanced Audio Codec (AAC) specifies a bandwidth voice codec that provides improved voice fidelity. This codec also provides equal or improved sound quality over older codecs with lower bit rates. AAC includes the following features:

- For AAC calls, the codec specifies Media_Payload_AAC 42.
- The maxFramesPerPacket specifies 1.
- Internet Low Bit Rate Codec (iLBC) enables graceful speech quality degradation in a lossy network where frames get lost. For iLBC calls, the codec specifies Media_Payload_ILBC = 86.

The system adds an audio bandwidth field to the CDR for AAC and iLBC calls.

Field Names	Definitions
origMediaCap_bandwidth	This integer field contains the audio bandwidth.
destMediaCap_bandwidth	This integer field contains the audio bandwidth.

The system populates the bandwidth fields based on the following table.

Codec	Bandwidth
G711Alaw64k	64
G711Alaw56k	56
G711Ulaw64k	64
G711Ulaw56k	56
G722_64k	64
G722_56k	56
G722_48k	48
G7231	7
G728	16
G729	8
G729AnnexA	8
G729AnnexB	8
G729AnnexAwAnnexB	8
XV150_MR_729A	8

NSE_VBD_729A	8
GSM_Full_Rate	13
GSM-Half_Rate	7
GSM_Enhanced_Full_Rate	13
Wide_Band_256k	256
Is11172AudioCap	0
Is13818AudioCap	0
Data64	64
Data56	56
GSM	13
G7221_32K	32
G7221_24K	24
AAC	256
ILBC	15k or 13k

AAC Calls CDR Example

The following table contains an example CDR for a call with AAC codec.

Calling party 51234 calls the called party 57890.

Global Call ID callid	Orig Leg Call Identifier	Dest Leg Call Identifier	Calling Party Number	Orig Called Party Number	Final Called Party Number	Last RedirectDn	Orig Cause Values	Dest Cause Value	Orig MediaCap Payload Capability	Orig MediaCap Bandwidth	Dest MediaCap Payload Capability	Dest MediaCap Bandwidth
121	101	102	51234	57890	57890	57890	0	16	42	256	42	256

iLBC Calls CDR Example

The following table contains an example CDR for a call with iLBC codec.

Calling party 51234 calls the called party 57890.

Global Call ID callid	Orig Leg Call Identifier	Dest Leg Call Identifier	Calling Party Number	Orig Called Party Number	Final Called Party Number	Last RedirectDn	Orig Cause Values	Dest Cause Value	Orig MediaCap Payload Capability
121	101	102	51234	57890	57890	57890	0	16	86

Orig MediaCap Bandwidth	Dest MediaCap Payload Capability	Dest MediaCap Bandwidth
15	86	15

Mobility

The system supports the following Mobility features:

- Hand-In
- Hand-Out
- Cell Pickup
- Interactive Voice Response (IVR)

The system generates a standard CDR for every call that uses the Mobility feature. When a call is split, redirected, or joined by the Mobility feature, the corresponding OnBehalfOf code represents a new value that is designated to the Mobility feature. The CAR Loader checks the following OnBehalfOf fields:

- origCallTerminationOnBehalfOf
- destCallTerminationOnBehalfOf
- origCalledPartyRedirectOnBehalfOf
- lastRedirectRedirectOnBehalfOf
- joinOnBehalfOf

If any of the preceding OnBehalfOf codes has the Mobility code of 24, the CDR has the Mobility call type that is determined by the CAR Loader, four redirectResource codes apply for Mobility features, including Hand-In (code 303), Hand-Out (code 319), Cell Pickup (code 335), and IVR (code 399).

Mobility CDR Examples

A dual-mode phone with Enterprise number of 22285 and a cell number of 9728324124 exists. The following table contains example CDRs for mobility calls that use the dual-mode phone in the following scenarios:

- Example A—Mobility Follow Me: 22202 calls 22285, and both 22285 and 9728324124 ring. The cell phone answers the call. The parties talk for 80 seconds.
- Example B—Mobility HandIn: A call goes to the cell phone. The parties talk for 39 seconds; the dual-mode phone gets carried into the Enterprise network, and the call gets switched from the cell network to the Enterprise network. The call lasts for another 15 seconds.
- Example C—Mobility HandOut: The handout number (H-number) specifies 555123. A call gets made to the Enterprise number 22285. They talk for 21 seconds; the dual-mode phone then gets carried out of the Enterprise network and into the cell network. The call gets switched from the Enterprise network to the cell network (9728324124). The call lasts another 39 seconds.
- Example D—Mobility Cell Pickup: A call gets made and established to 22285. They talk for 40 seconds; then, Cell Pickup gets invoked. The call gets switched from the Enterprise phone to the cell phone. The call continues for another 111 seconds.

- Example E—Mobility IVR: A call comes into the Cisco Unified Communications Manager with a string (DID#RemoteDest#TargetNum#). The call gets redirected to the TargetNum. 9728131234 calls into an IVR, and data gets collected. The target destination specifies 812345 and the call gets redirected to 812345. The call connects for 60 seconds.

	Global Call ID callid	Orig Leg Call Identifier	Dest Leg Call Identifier	Calling Party Number	Orig Called Party Number	Final Called Party Number	Last RedirectDn	Orig Cause Values
A—Follow Me Call CDR	861	22481077	22481078	22202	22285	9728324124	22285	16
B—Mobility HandIn - Call to cell #9728324214 CDR	864	22481083	22481085	22202	919728324124	919728324124	9199728324124	393216
B—HandIn Call to the Enterprise CDR	864	22481083	22481087	22202	22285	22285	22285	0
C—HandOut Enterprise Call to 22285 CDR	964	22481093	22481094	22202	22285	22285	22285	393216
C—HandOut Server Call from Cell Phone to H-Number CDR	965	22481095	22481096	9728324124	555123	555123	555123	393216
C—HandOut Call CDR	964	22481093	22481095	22202	9728324124	9728324124	9728324124	0
D—Mobility Cell Pickup Enterprise Call to 22285 CDR	555	22481111	22481112	22202	22285	22285	22285	393216
D—Mobility Cell Pickup Server Call to Cell Phone CDR	566	22481222	22481223		9728324124	9728324124	9728324124	0
D—Mobility Final Handout Call CDR	964	22481111	22481222	22202	9728324124	9728324124	0728324124	0
E—Mobility IVR CDR	12345	16677100	16677102	9728131234	8005559876	812345	8005559876	0

Dest Cause Value	Last Redirect Redirect Reason	Last Redirect Redirect OnBehalfOf	Orig Termination OnBehalfOf	Dest Termination OnBehalfOf	Joint OnBehalfOf	Duration
0	0	0			0	80
393216	0	0	24	24	0	39
16	303	24	24	12	24	15

Dest Cause Value	Last Redirect Redirect Reason	Last Redirect Redirect OnBehalfOf	Orig Termination OnBehalfOf	Dest Termination OnBehalfOf	Joint OnBehalfOf	Duration
393216	0	0	24	24	0	21
393216	0	0	24	24	0	0
16	319	24	24	12	24	39
393216	0	0	24	24	0	40
0	0	0	24	24	0	0
16	335	24	24	12	24	111
16	399	24	0	0	N/A	60

Intercom

The Intercom feature provides one-way audio; therefore, the CDR reflects one-way audio. For talk-back intercom, two-way audio exists, and the CDR reflects two-way audio.

The Intercom feature requires a partition (intercom partition) and existing CDR partition fields get used to identify intercom calls.

Intercom CDR Example

Phone 20000 invokes the intercom in the following scenarios:

- Example A—Whisper Intercom: The configured intercom partition specifies "intercom."
- Example B—Talk-Back Intercom: Phone 20000 presses the Intercom button. 20001 invokes talk-back and talks to 20000. The configured intercom partition specifies "intercom."

	Global Call ID callid	Orig Leg Call Identifier	Dest Leg Call Identifier	Calling Party Number	Orig Called Party Number	Final Called Party Number	Orig Cause Values	Dest Cause Value
A—Whisper Intercom CDR	1111000	21822467	21822468	20000	20001	20001	16	0
B—Talk-Back Intercom CDR	1111000	21822469	21822470	20000	20001	20001	16	0T

Orig Media Transport Address IP	Orig Media Transport Address Port	Dest Media Transport Address IP	Dest Media Transport Address Port	Orig Called Party Number Partition	Calling Party Number Partition	Final Called Party Number Partition	Duration
0	0	-47446006	28480	Intercom	Intercom	Intercom	5
-131332086	29458	-47446006	29164	Intercom	Intercom	Intercom	5

Original Calling Party on Transfer

This feature changes the calling party number for a consultation call of a Cisco Unity Connection-initiated call transfer. The CDR of the consultation call shows that the original caller calls the transfer destination, not that the Cisco Unity Connection port calls the transfer destination.

You must configure this feature in the service parameters in Cisco Unified Communications Manager. See additional information at "Configuring CDR Service Parameters" section on page 2-2.

Original Calling Party on Transfer CDR Example

4001 calls 4002. 4002 transfers the call to 4003. The system generates three CDRs:

- The call between the original parties (4001 to 4002).
- The consultation call between the transferring party (4002) to the final transfer destination (4003).
- The call from the transferred party (4001) to the transfer destination (4003).

Call	CallingPartyNumber	originalCalledPartyNumber
1	4001	4002
2	4002	4003
3	4001	4003

Note No originalCallingParty field exists in the CDR.

Interpreting Cisco Personal Assistant Data in the CDRs

The Cisco Personal Assistant application can selectively handle incoming calls and assist with outgoing calls. This section provides a brief overview of personal assistant and describes the personal assistant call types with example CDR scenarios.

Personal assistant provides the following features:

- **Rule-Based Call Routing**—Personal assistant can forward and screen incoming calls based on rules that users devise. Personal assistant can handle incoming calls according to caller ID, date and time of day, or the user meeting status based on the user calendar (such as office hours, meeting schedules, vacations, holidays, and so forth). Personal assistant can also selectively route calls to other telephone numbers.

  Thus, personal assistant can route an incoming call to a desk phone, to a cell phone, home phone, or other phone, based on the call routing rules that users create. An incoming call can even generate an e-mail-based page.
- **Speech-Enabled Directory Dialing**—Personal assistant allows users to dial a phone number by speaking the name of the called person. Personal assistant then obtains the telephone number of that person from the corporate directory or personal address book.
- **Speech-Enabled Voice-Mail Browsing**—Users can use voice commands to browse, listen to, and delete voice-mail messages.
- **Speech-Enabled Simple Ad Hoc Conferencing**—Users can initiate conference calls by telling personal assistant to set up a conference call with the desired participants.

Personal assistant provides the following call types:

Personal Assistant Direct Call

A personal assistant direct call acts similar to the Transfer without Consultation call type. See the "Transfer Without Consultation" section on page 10-14.

Personal Assistant Direct Call CDR Example

The following table contains an example CDR for this scenario:

- User A (2101) calls Personal Assistant route point (2000) and says "call User B."
- The call transfers to User B (2105). In this case, User B did not configure any rules.

Note In the following example, 2000 represents the main personal assistant route point to reach personal assistant, 21XX represents the personal assistant interceptor route point, and 2001 - 2004 represents the media port.

In all cases, 2101 specifies the calling number.

Calling Party Num	Orig LegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Num	Final Called Party Number Partition	Original Called Party Num	Original Called Party Number Partition	Last Redir DN	Last Redirect DN Partition	Duration (secs)
2101	16777217	PAManaged	16777219	2004	Phones	2000	1023970182	2000	Phones	34
2004	16777221	Phones	16777222	2105	PAManaged	2105	1023970182	2105	PAManaged	0
2101	16777217	PAManaged	16777222	2105	PAManaged	2105	1023970191	2105	PAManaged	5

Personal Assistant Interceptor Going to Media Port and Transferring the Call

This scenario acts similar to Transfer without Consultation and Forwarded Calls actions. See the sections on "Transfer Without Consultation" section on page 10-14 and "Forwarded or Redirected Calls" section on page 10-11.

Personal Assistant Interceptor Going to Media Port and Transferring the Call CDR Example

The following table contains an example CDR for this scenario:

- User A (2101) dials 2105.
- The personal assistant interceptor (21XX) picks up the call and redirects it to a media port (2002).
- Personal assistant processes the call according to the rules (if any) and transfers the call to the destination (2105), which has not configured any rules.

.

Calling Party Num	Orig LegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Num	Final Called Party Number Partition	Original Called Party Num	Original Called Party Number Partition	Last Redir DN	Last Redirect DN Partition	Duration (secs)
2002	16777234	Phones	16777285	2105	PAManaged	2105	1023970478	2105	PAManaged	2
2101	16777230	PAManaged	16777232	2002	PA	2105	1023970478	21xx	“ “	9
2105	16777235	PAManaged	16777230	2101	“ “	“ “	1023970483	“ “	“ “	5

Personal Assistant Interceptor Going Directly to Destination

This scenario can have two different cases: with no rules and with rules.

Personal Assistant Going Directly to Destination with No Rules CDR Example

The following table contains an example CDR for this scenario:

- User A (2101) dials 2105.
- The personal assistant interceptor (21XX) picks up the call, processes it according to the rules (if any), and redirects the call to the destination (2105).

The following table contains an example CDR for this scenario:

Calling Party Number	OrigLeg Call Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number	Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration (secs)
2101	16777240	PAManaged	16777242	2105	PA	2105	1023970710	21XX	“ “	8

Personal Assistant Going Directly to Destination with Rule to Forward Calls to a Different Destination CDR Example

The following table contains an example CDR for this scenario:

- User A (2101) dials 2105.
- The Personal Assistant interceptor (21XX) picks up the call and processes it according to the rules.
- The Personal Assistant interceptor then redirects the call to the final destination (2110). In this case, 2105 configured a rule to forward the call to extension 2110.

Calling Party Number	OrigLeg Call Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number	Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration (secs)
2101	16777240	PAManaged	16777242	2110	PA	2105	1023970710	21XX	“ “	8

Personal Assistant Interceptor Going to Multiple Destinations

This scenario can have several different cases. In each case, User B (2105) configured a rule to reach him at extension 2110 or 2120. This rule could activate when a caller calls Personal Assistant route point (2000) and says “call User B” (direct case) or when the caller dials User B (2105) directly (interceptor case).

Personal Assistant Interceptor Going to Multiple Destinations CDR Examples

The following sections contain examples of each case. The tables contain example CDRs for each of these scenarios:

- Personal Assistant Direct Multiple Destinations: 2110 and 2120 (Call Accepted at First Destination), page 10-33
- Personal Assistant Direct Multiple Destinations: 2110 and 2120 (Call Accepted at Second Destination), page 10-33
- Personal Assistant Direct Multiple Destinations: 2110 and 2120 (Call Accepted at Third Destination), page 10-34
- Personal Assistant Intercept Multiple Destinations: 2110 and 2120 (Call Accepted at First Destination), page 10-34
- Personal Assistant Intercept Multiple Destinations: 2110 and 2120 (Call Accepted at Second Destination), page 10-35
- Personal Assistant Intercept Multiple Destinations: 2110 and 2120 (Call Accepted at Third Destination), page 10-35

Personal Assistant Direct Multiple Destinations: 2110 and 2120 (Call Accepted at First Destination)

- User A calls personal assistant and says “call User B.”
- User B answers the call at 2110 extension.

Calling Party Num	Orig LegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Num	Final Called Party Number Partition	Original Called Party Num	Original Called Party Number Partition	Last Redir DN	Last Redirect DN Partition	Duration (secs)
2004	16777262	Phones	16777263	2110	PAManaged	2110	1023971303	2110	PAManaged	6
2101	16777258	PAManaged	16777260	2004	Phones	2000	1023971303	2000	Phones	22
2110	16777263	PAManaged	16777258	2101	“ “	“ “	1023971312	“ “	“ “	9

Personal Assistant Direct Multiple Destinations: 2110 and 2120 (Call Accepted at Second Destination)

- User A calls personal assistant and says “call User B.”
- User B answers the call at 2120 extension.

Calling Party Num	Orig LegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Num	Final Called Party Number Partition	Original Called Party Num	Original Called Party Number Partition	Last Redir DN	Last Redirect DN Partition	Duration (secs)
2001	16777269	Phones	16777270	2110	PAManaged	2110	1023971456	2110	PAManaged	0
2001	16777272	Phones	16777273	2120	PAManaged	2120	1023971467	2120	PAManaged	4
2101	16777265	PAManaged	16777267	2001	Phones	2000	1023971467	2000	Phones	37
2120	16777273	PAManaged	16777265	2101	“ “	“ “	1023971474	“ “	“ “	7
2110	16777275	PAManaged	0	“ “	“ “	“ “	1023971476	“ “	“ “	0

Personal Assistant Direct Multiple Destinations: 2110 and 2120 (Call Accepted at Third Destination)

- User A calls personal assistant and says “call User B.”
- User B does not answer at either extension 2110 or 2120.
- Personal Assistant transfers the call to the original destination (2105), and User B then answers at that extension.

Note 2105 (the original destination) represents the third destination in this case.

Calling Party Num	Orig LegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Num	Final Called Party Number Partition	Original Called Party Num	Original Called Party Number Partition	Last Redir DN	Last Redirect DN Partition	Duration (secs)
2002	16777281	Phones	16777282	2110	PAManaged	2110	1023971602	2110	PAManaged	0
2002	16777284	Phones	16777285	2120	PAManaged	2120	1023971615	2120	PAManaged	0
2101	16777277	PAManaged	16777279	2002	Phones	2000	1023971619	2000	Phones	38
2002	16777287	Phones	16777288	2105	PAManaged	2105	1023971619	2105	PAManaged	0
2101	16777277	PAManaged	16777288	2105	PAManaged	2105	1023971627	2105	PAManaged	7
2105	16777289	PAManaged	0	“ “	“ “	“ “	1023971629	“ “	“ “	0

Personal Assistant Intercept Multiple Destinations: 2110 and 2120 (Call Accepted at First Destination)

- User A calls personal assistant and says “call User B.”
- User B answers the call at extension 2110.

Calling Party Num	Orig LegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Num	Final Called Party Number Partition	Original Called Party Num	Original Called Party Number Partition	Last Redir DN	Last Redirect DN Partition	Duration (secs)
2003	16777295	Phones	16777296	2110	PAManaged	2110	1023971740	2110	PAManaged	4

Calling Party Num	Orig LegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Num	Final Called Party Number Partition	Original Called Party Num	Original Called Party Number Partition	Last Redir DN	Last Redirect DN Partition	Duration (secs)
2101	16777291	PAManaged	16777293	2003	PA	2105	1023971740	21XX	“ “	10
2110	16777296	PAManaged	16777291	2101	“ “	“ “	1023971749	“ “	“ “	9

Personal Assistant Intercept Multiple Destinations: 2110 and 2120 (Call Accepted at Second Destination)

- User A calls personal assistant and says “call User B.”
- User B answers the call at extension 2120.

Calling Party Num	Orig LegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Num	Final Called Party Number Partition	Original Called Party Num	Original Called Party Number Partition	Last Redir DN	Last Redirect DN Partition	Duration (secs)
2004	16777302	Phones	16777303	2110	PAManaged	2110	1023971815	2110	PAManaged	0
2004	16777305	Phones	16777306	2120	PAManaged	2120	1023971824	2120	PAManaged	3
2101	16777298	PAManaged	16777300	2004	PA	2105	1023971824	21XX	“ “	22
2120	16777306	PAManaged	16777298	2101	“ “	“ “	1023971832	“ “	“ “	8

Personal Assistant Intercept Multiple Destinations: 2110 and 2120 (Call Accepted at Third Destination)

- User A calls personal assistant and says “call User B.”
- User B does not answer at either extension 2110 or 2120.
- Personal assistant transfers the call to the original destination (2105), which User B then answers.

Note 2110 (the original destination) represents the third destination in this case.

Calling Party Num	Orig LegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Num	Final Called Party Number Partition	Original Called Party Num	Original Called Party Number Partition	Last Redir DN	Last Redirect DN Partition	Duration (secs)
2001	16777312	Phones	16777313	2110	PAManaged	2110	1023971923	2110	PAManaged	0
2001	16777315	Phones	16777316	2120	PAManaged	2120	1023971936	2120	PAManaged	0
2101	16777308	PAManaged	16777310	2001	PA	2105	1023971940	21XX	“ “	30
2001	16777318	Phones	16777319	2105	PAManaged	2105	1023971940	2105	PAManaged	0
2101	16777308	PAManaged	16777319	2105	PAManaged	2105	1023971953	2105	PAManaged	12

Personal Assistant Conferencing

Personal assistant conferencing acts similar to the ad hoc conferences call type. For more information, see the "Conference Calls" section on page 10-15.

Personal Assistant Conferencing CDR Example

The following table contains an example CDR for this scenario:

- User A calls personal assistant route point (2000) and says "conference User B (2105) and User C (2110)."
- Personal assistant conferences User B and C into User A conference.

Calling Party Num	Orig LegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Num	Final Called Party Number Partition
2003	16777345	Phones	16777346	2105	PAManaged
2101	16777340	PAManaged	16777342	2003	Phones
2003	16777350	Phones	16777351	2002	PAManaged
2003	16777342	Phones	16777347	2110	" "
2110	16777351	PAManaged	16777352	b00110201001	" "
2105	16777346	PAManaged	16777349	b00110201001	" "
2101	16777340	PAManaged	16777348	b00110201001	" "

This table continues with this additional information.

Original Called Party Number	Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration (seconds)
2105	1023972575	2105	PAManaged	6
2000	1023972576	2003	Phones	62
2110	1023972595	2110	PAManaged	39
b00110201001	1023972601	b00110201001	" "	25
b00110201001	1023972609	b00110201001	" "	14
b00110201001	1023972610	b00110201001	" "	34
b00110201001	1023972610	b00110201001	" "	34

Call Scenarios

Each normal call between two parties logs one CDR. Each CDR contains all fields that are identified in preceding scenarios, but some fields may not be used. If a field is not used, it stays blank if it is an ASCII string field or shows "0" if it is a numeric field. When supplementary services are involved in a call, more CDRs may get written.

In addition to the CDR, one CMR per endpoint may get involved in a call. In a normal call between two parties each using an IP phone, two CMRs get written, one for the originator and one for the destination of the call.

This section describes the records that are written for different call types, including all records for each call and important fields that are shown in summary tables for easy viewing and comparison.

- Normal Calls (IP Phone to IP Phone), page 10-38
- Abandoned Calls, page 10-38
- Calls With Busy or Bad Destinations (Unsuccessful Calls), page 10-40
- Forwarded Calls, page 10-41
- Call Pickup, page 10-42
- Call Pickup, page 10-44
- Transferred Calls, page 10-45
- Conference Calls, page 10-48
- Secure Meet-Me Conference, page 10-50
- Ad Hoc Conference Linking, page 10-50
- Call Park, page 10-61
- Precedence Calls (MLPP), page 10-63
- Malicious Calls, page 10-64
- Immediate Divert (to Voicemail), page 10-64
- Barge, page 10-66
- cBarge, page 10-68
- Video Calls, page 10-69
- Forced Authorization Code (FAC), page 10-70
- Client Matter Code (CMC), page 10-70
- Call Secured Status, page 10-71
- Secure Meet-Me Conference, page 10-50
- DTMF Method, page 10-72
- RSVP, page 10-73
- Redirection (3xx) Calls, page 10-74
- Replaces Calls, page 10-75
- Refer Calls, page 10-76
- Monitor Calls, page 10-76
- Recording Calls, page 10-77
- AAC and iLBC Calls, page 10-79
- Mobility, page 10-80
- Intercom Calls, page 10-83

Normal Calls (IP Phone to IP Phone)

Normal calls log three records per call; one CDR and two CMRs, one for each endpoint. In the CDR, the "originalCalledPartyNumber" field contains the same Directory Number as the "finalCalledPartyNumber" field.

Examples of Successful Calls

A successful call between two Cisco Unified IP Phones generates a single CDR at the end of the call.

- The caller terminated a 60-second call. Because the calling party hangs up, the orig_CauseValue specifies 16 (Normal Clearing).

Field Names	CDR
globalCallID_callId	1
origLegCallIdentifier	100
destLegCallIdentifier	101
callingPartyNumber	2001
originalCalledPartyNumber	2309
finalCalledPartyNumber	2309
lastRedirectDn	2309
origCause_Value	16
dest_CauseValue	0
duration	60

- The called party clears a 60-second call. Because the called party hangs up, the dest_CauseValue specifies 16 (Normal Clearing).

Field Names	CDR
globalCallID_callId	1
origLegCallIdentifier	100
destLegCallIdentifier	101
callingPartyNumber	2001
originalCalledPartyNumber	2309
finalCalledPartyNumber	2309
lastRedirectDn	2309
origCause_Value	0
dest_CauseValue	16
duration	60

Abandoned Calls

Be aware that the logging of calls with zero duration is optional. Normally these records do not get logged. If logging calls with zero duration is enabled, all calls will generate a CDR.

If the call was abandoned, such as when a phone is taken off hook and placed back on hook, various fields will not contain data. In this case, the **originalCalledPartyNumber**, **finalCalledPartyNumber**, the partitions associated with them, **destIpAddr**, and the **dateTimeConnect** fields stay blank. All calls that were not connected will have a **duration** of zero seconds. When a call is abandoned, the cause code specifies “0”.

If the user dialed a Directory Number and then abandoned the call before it was connected, the origCalledPartyNumber and finalcalledPartyNumber fields and their associated partitions contain the directory number and partition to which the call would have been extended. The **destIPAddress** field stays blank and the **duration** specifies zero.

Examples of Abandoned Calls

- Extension 2001 goes off hook, then on hook.

Field Names	CDR
globalCallID_callId	1
origLegCallIdentifier	100
destLegCallIdentifier	0
callingPartyNumber	2001
originalCalledPartyNumber	
finalCalledPartyNumber	
lastRedirectDn	
origCause_Value	16
dest_CauseValue	0
duration	0

- Extension 2001 calls 2309, but 2001 hangs up (abandons) the call before it is answered.

Field Names	CDR
globalCallID_callId	2
origLegCallIdentifier	200
destLegCallIdentifier	201
callingPartyNumber	2001
originalCalledPartyNumber	2309
finalCalledPartyNumber	2309
lastRedirectDn	2309
origCause_Value	16
dest_CauseValue	0
duration	0

Calls With Busy or Bad Destinations (Unsuccessful Calls)

These calls will all get logged as a normal call with all relevant fields that contain data. The Calling or Called Party Cause field contains a cause code that indicates why the call was not connected, and the Called Party IP and Date/Time Connect fields stay blank. All unsuccessful calls get logged, even if zero duration calls are not being logged.

Examples of Unsuccessful Calls

- Call to PSTN number, party engaged (cause 17 = user busy)

Field Names	CDR
globalCallID_callId	3
origLegCallIdentifier	300
destLegCallIdentifier	301
callingPartyNumber	2001
originalCalledPartyNumber	9728134987
origCause_Value	0
dest_CauseValue	17
duration	0

- Call to PSTN number, number does not exist (cause 1 = number unavailable)

Field Names	CDR
globalCallID_callId	4
origLegCallIdentifier	302
destLegCallIdentifier	303
callingPartyNumber	2001
originalCalledPartyNumber	9728134987
origCause_Value	1
dest_CauseValue	0
duration	0

- Call to PSTN fails because PSTN trunks are out of order (cause 38 = Network Out Of Order).

Field Names	CDR
globalCallID_callId	5
origLegCallIdentifier	304
destLegCallIdentifier	305
callingPartyNumber	2001
originalCalledPartyNumber	9728134987
origCause_Value	0
dest_CauseValue	38
duration	0

Forwarded Calls

Call Forwarding uses the redirect call primitive to forward the call. Features that use the redirect call primitive will have similar CDRs. The following list gives some of the important CDR fields for forwarded calls.

- The **originalCalledPartyNumber** contains the number of the original called party.
- The **finalCalledPartyNumber** specifies the number that answered the call.
- The **lastRedirectDn** field specifies the number that performed the last redirect.
- The **origCalledPartyRedirectReason** specifies the reason that the call was redirected the first time. For call forwarding, this field can contain (**Call Forward Busy=1, Call Forward No Answer=2, Call Forward All=15**).
- The **lastRedirectRedirectReason** specifies the reason that the call was redirected the last time. For call forwarding, this field can contain (**Call Forward Busy=1, Call Forward No Answer=2, Call Forward All=15**).
- The **origCalledPartyRedirectOnBehalfOf** field identifies which feature redirects the call for the first redirect. For call forwarding, this field specifies 5 (Call Forward).
- The **lastRedirectRedirectOnBehalfOf** field identifies which feature redirects the call for the last redirect. For call forwarding, this field specifies 5 (Call Forward).

Forwarding Examples

- **CFA Example** - Call comes in from the PSTN to extension 2001, the call gets forwarded (CFA) to 2309, where the call is answered, and the call duration is 2 minutes.

Field Names	CDR
globalCallID_callId	12345
origLegCallIdentifier	100
destLegCallIdentifier	102
callingPartyNumber	9728134987
originalCalledPartyNumber	2001
finalCalledPartyNumber	2309
lastRedirectDn	2001
origCause_Value	0
dest_CauseValue	16
origCalledPartyRedirectReason	15
lastRedirectRedirectReason	15
origCalledPartyRedirectOnBehalfOf	5
lastRedirectRedirectOnBehalfOf	5
duration	120

- **Multiple Hop CFA & CFNA Example** - Call comes in from the PSTN to extension 1000, the call gets forwarded (CFA) to 2000; then, the call gets forwarded (CFNA) to voice mail (6000) where the caller leaves a message.

Field Names	CDR
globalCallID_callId	12346
origLegCallIdentifier	102
destLegCallIdentifier	105
callingPartyNumber	9728134987
originalCalledPartyNumber	1000
finalCalledPartyNumber	6000
lastRedirectDn	2000
origCause_Value	0
dest_CauseValue	16
origCalledPartyRedirectReason	15
lastRedirectRedirectReason	2
origCalledPartyRedirectOnBehalfOf	5
lastRedirectRedirectOnBehalfOf	5
duration	15

- **Multiple Hop CFNA & CFB Example** - Call comes in from the PSTN to extension 4444, the call gets forwarded (CFNA) to 5555; then, it gets forwarded (CFB) to 6666 where the call is answered and they talk for 30 seconds.

Field Names	CDR
globalCallID_callId	12347
origLegCallIdentifier	106
destLegCallIdentifier	108
callingPartyNumber	9728134987
originalCalledPartyNumber	4444
finalCalledPartyNumber	6666
lastRedirectDn	5555
origCause_Value	16
dest_CauseValue	0
origCalledPartyRedirectReason	2
lastRedirectRedirectReason	1
origCalledPartyRedirectOnBehalfOf	5
lastRedirectRedirectOnBehalfOf	5
duration	30

Call Pickup

The following two types of call pickup in Cisco Unified Communications Manager exist:

- Pickup, page 10-43
- Auto Pickup, page 10-43

The CDRs differ slightly for each type of call pickup.

Pickup

Pickup Call Example

Call comes in from the PSTN to extensions 2000, 2001, and 2002, which are in the same pickup group. Extension 2002 picks up the call that is ringing on 2001. Extension 2002 answers the call, and the call connects between the PSTN caller and extension 2002.

Field Names	Pickup Call CDR
globalCallID_callId	22
callingPartyNumber	9728131234
originalCalledPartyNumber	2001
finalCalledPartyNumber	2002
lastRedirectDn	2001
origCause_Value	0
dest_CauseValue	16
origTerminationOnBehalfOf	16
destTerminationOnBehalfOf	16
lastRedirectOnBehalfOf	16
lastRedirectReason	5
joinOnBehalfOf	16

Auto Pickup

Auto Pickup acts like call pickup with auto answer. It does not require the last answer softkey press. The call automatically connects. Two CDRs get generated for Auto Pickup. These CDR will have the same Call ID.

- The first CDR gets generated for the original call. This CDR will have the **origTerminationOnBehalfOf** and **destTerminationOnBehalfOf** fields equal to 16 (Pickup). This indicates that the call was terminated on behalf of the Pickup feature.
- The second CDR represents the final call after it was picked up. This CDR will have the **lastRedirectOnBehalfOf** and the **joinOnBehalfOf** fields set to 16 (Pickup). This indicates that the call was joined on behalf of the Pickup feature. The **lastRedirectReason** contains the redirect reason of 5 (Pickup).

Auto Pickup CDRs will look the same for all types of auto pickup: Auto Pickup, Auto Group Pickup and Auto Other Pickup.

Auto Pickup Example

- **Auto Pickup Example** - Call from the PSTN to extension 2001. 2001 and 2002 exist in the same pickup group. 2002 picks up the call that is ringing on 2001; the call automatically connects between the PSTN caller and 2002. They talk for 2 minutes.

Field Names	Original Call CDR	Pickup CDR
globalCallID_callId	11	11
origLegCallIdentifier	12345	12345
destLegCallIdentifier	12346	12347
callingPartyNumber	9728134987	9728134987
originalCalledPartyNumber	2001	2002
finalCalledPartyNumber	2001	2002
lastRedirectDn	2001	2001
origCause_Value	393216	16
dest_CauseValue	393216	0
origTerminationOnBehalfOf	16	12
destTerminationOnBehalfOf	16	16
lastRedirectRedirectReason	0	5
lastRedirectRedirectOnBehalfOf	0	16
joinOnBehalfOf	0	16
duration	0	120

Call Pickup

Legacy Call Pickup calls act very similar to forwarded calls. Legacy Call Pickup uses the redirect call control primitive just like call forwarding. The following list gives the important CDR fields for Legacy Call Pickup calls.

- The **originalCallPartyNumber** contains the number of the original called party.
- The **finalCalledPartyNumber** specifies the number of the party that picked up the call.
- The **lastRedirectDn** field specifies the number that was ringing when the call was picked up.
- The **origCalledPartyRedirectReason** specifies the reason that the call was redirected the first time. For call pickup calls this field can contain (**Call Pickup = 5**).
- The **lastRedirectRedirectReason** specifies the reason that the call was redirected the last time. For call pickup, this field can contain (**Call Pickup = 5**).
- The **origCalledPartyRedirectOnBehalfOf** field identifies which feature redirects the call for the first redirect. For call pickup, this field specifies (**Pickup = 16**).
- The **lastRedirectRedirectOnBehalfOf** field identifies which feature redirects the call for the last redirect. For call pickup, this field specifies (**Pickup = 16**).

Legacy Call Pickup Example

Call from the PSTN to extension 2001, 2001 and 2002 exist in the same pickup group. 2002 picks up the call ringing on 2001, 2002 answers the call, and the call connects between the PSTN caller and 2002. They talk for 2 minutes.

Field Names	CDR
globalCallID_callId	22
origLegCallIdentifier	1
destLegCallIdentifier	2
callingPartyNumber	9728134987
originalCalledPartyNumber	2001
finalCalledPartyNumber	2002
lastRedirectDn	2001
origCause_Value	0
dest_CauseValue	16
origCalledPartyRedirectReason	0
lastRedirectRedirectReason	5
origCalledPartyRedirectOnBehalfOf	16
lastRedirectRedirectOnBehalfOf	16
duration	120

Transferred Calls

Calls that are transferred generate multiple CDRs. One CDR occurs for the original call, one for the consultation call, and another for the final transferred call.

For the original call, the **origCause_value** and **destCause_value** gets set to (split = 393216), which indicates the call was split. The **origCallTerminationOnBehalfOf** and **destCallTerminationOnBehalfOf** fields get set to (Transfer = 10) to indicate that this call was involved in a transfer.

For the consultation call, the **origCause_value** and **destCause_value** gets set to (split = 393216), which indicates that the call was split. The **origCallTerminationOnBehalfOf** and **destCallTerminationOnBehalfOf** fields get set to (Transfer = 10) to indicate that this call was involved in a transfer.

For the final transferred call, the **joinOnBehalfOf** field gets set to (Transfer = 10) to indicate that this call resulted from a transfer.

Transfer Examples

The following examples do not represent an exhaustive set and are intended to illustrate the records that would be generated under the stated circumstances. These examples help clarify what records are generated on transferred calls.

- **Blind Transfer from the Calling Party** - Call from extension 2001 to a PSTN number, they talk for 120 seconds. 2001 initiates a blind transfer to 2002. **CDR 1** (original call) shows a call from extension 2001 to a PSTN number, talking for 120 seconds. **CDR 2** (consultation call) shows a call from 2001 to extension 2002. **CDR 3** is the final transferred call where 2001 completes the transfer, drops out of the call, leaving a call between the PSTN and 2002.

Field Names	Original Call CDR	Consultation Call CDR	Final Transferred CDR
globalCallID_callId	1	2	1
origLegCallIdentifier	101	103	102
destLegCallIdentifier	102	104	104
callingPartyNumber	2001	2001	3071111
originalCalledPartyNumber	3071111	2002	2002
finalCalledPartyNumber	3071111	2002	2002
lastRedirectDn	3071111	2002	2001
origCause_Value	393216	393216	16
dest_CauseValue	393216	393216	0
origTerminationOnBehalfOf	10	10	0
destTerminationOnBehalfOf	10	10	0
joinOnBehalfOf	0	0	10
duration	120	0	360

- **Consultation Transfer from the Calling Party** - Call from extension 2001 to a PSTN number; they talk for 60 seconds. 2001 initiates a consultation transfer to 2002 and talks for 10 seconds before the transfer completes. The final transferred call talks for 360 seconds. **CDR 1** (original call) shows a call from extension 2001 to a PSTN number, talking for 60 seconds. **CDR 2** (consultation call) shows a call from 2001 to extension 2002, talking for 10 seconds. **CDR 3** represents the final transferred call where 2001 completes the transfer, drops out of the call, leaving a call between the PSTN and 2002.

Field Names	Original Call CDR	Consultation Call CDR	Final Transferred Call CDR
globalCallID_callId	1	2	1
origLegCallIdentifier	111	113	112
destLegCallIdentifier	112	114	114
callingPartyNumber	2001	2001	3071111
originalCalledPartyNumber	3071111	2002	2002
finalCalledPartyNumber	3071111	2002	2002
lastRedirectDn	50001	50001	2001
origCause_Value	393216	393216	16
dest_CauseValue	393216	393216	0
origTerminationOnBehalfOf	10	10	0
destTerminationOnBehalfOf	10	10	0

joinOnBehalfOf	0	0	10
duration	60	10	360

- **Blind Transfer from the Called Party** - Call from 50000 to 50001, they talk for 120 seconds. 50001 initiates a blind transfer to 50002. **CDR 1** (original call) shows a call from extension 50001 to 50002, talking for 120 seconds. **CDR 2** (consultation call) shows a call from 50001 to extension 50002. **CDR 3** represents the final transferred call where 50001 completes the transfer, drops out of the call, leaving a call between 50000 and 50002.

Field Names	Original Call CDR	Consultation Call CDR	Final Transferred Call CDR
globalCallID_callId	1	2	1
origLegCallIdentifier	200	202	200
destLegCallIdentifier	201	203	203
callingPartyNumber	50000	50001	50000
originalCalledPartyNumber	50001	50002	50002
finalCalledPartyNumber	50001	50002	50002
lastRedirectDn	50001	50001	50001
origCause_Value	393216	393216	16
dest_CauseValue	393216	393216	0
origTerminationOnBehalfOf	10	10	0
destTerminationOnBehalfOf	10	10	0
joinOnBehalfOf	0	0	10
duration	120	0	360

- **Consultation Transfer from the Called Party** - Call from 50000 to 50001, they talk for 120 seconds. 50000 initiates a blind transfer to 50002. **CDR 1** (original call) shows a call from extension 50000 to a 50001, talking for 120 seconds. **CDR 2** (consultation call) shows a call from 50000 to extension 50002. **CDR 3** represents the final transferred call where 50000 completes the transfer, drops out of the call, leaving a call between 50001 and 50002.

Field Names	Original Call CDR	Consultation Call CDR	Final Transferred Call CDR
globalCallID_callId	1	2	1
origLegCallIdentifier	200	202	201
destLegCallIdentifier	201	203	203
callingPartyNumber	50000	50001	50000
originalCalledPartyNumber	50001	50002	50002
finalCalledPartyNumber	50001	50002	50002
lastRedirectDn	50001	50001	50001
origCause_Value	393216	393216	16
dest_CauseValue	393216	393216	0
origTerminationOnBehalfOf	10	10	0

destTerminationOnBehalfOf	10	10	0
joinOnBehalfOf	0	0	10
duration	120	0	360

Conference Calls

Calls that are part of a conference have multiple records that are logged for them. The number of CDR records that are generated depends on the number of parties in the conference. One CDR exists for each party in the conference, one CDR for the original placed call, one CDR for each setup call that was used to join other parties to the conference, and one CDR for the last two parties that connected in the conference. Therefore, for a three-party ad-hoc conference, six CDRs would exist: one CDR for the original call, three CDRs for the parties that connected to the conference, one CDR for each setup call, and one CDR for the final two parties in the conference. You can associate the setup calls with the correct call leg in the conference by examining the calling leg ID and called leg ID.

The conference bridge device has special significance to the Cisco Unified Communications Manager, and calls to the conference bridge appear as calls to the conference bridge device. A special number in the form “b0019901001” shows the conference bridge port. All calls get shown “into” the conference bridge, regardless of the actually direction; however, by examining the setup call CDRs, you can determine the original direction of each call.

You can find the conference controller information in the comment field of the CDR. The format of this information follows:

Comment field = “ConfControllerDn=1000;ConfControllerDeviceName=SEP0003”

- The conference controller DN + conference controller device name uniquely identify the conference controller. The system needs the device name in the case of shared lines.
- If the call is involved in multiple conference calls, the comment field contains multiple conference controller information. This situation could occur when the conference goes down to two parties, and one of these parties starts another conference. If this is the case, the **last** conference controller information in the comment field will identify the conference controller.

The call legs connected to the conference will have the following fields information:

- The **finalCalledPartyNumber** field contains the conference bridge number “b0019901001”.
- The **origCalledPtyRedirectOnBehalfOf** field gets set to (Conference = 4).
  - The **lastRedirectRedirectOnBehalfOf** field gets set to (Conference = 4).
  - The **joinOnBehalfOf** field gets set to (Conference = 4).
  - The **comment** field identifies the conference controller.
  - The **destConversationID** field remains the same for all members in the conference. You can use this field to identify members of a conference call.

The original placed call and all setup calls that were used to join parties to the conference will have the following characteristics:

- The **origCallTerminationOnBehalfOf** field gets set to (Conference = 4).
- The **destCallTerminationOnBehalfOf** field gets set to (Conference = 4).

Conference Example

Call from 2001 to 2309.

2309 answers and talks for 60 seconds.

2001 presses the “conference” softkey and dials 3071111.

307111 answers and talks for 20 seconds, then 2001 presses the conference softkey to complete the conference.

The three members of the conference talk for 360 seconds.

3071111 hangs up, leaving 2001 and 2309 in the conference. Because there are only two participants left in the conference, the conference features joins these two directly together, and they talk for another 55 seconds.

Note Each conference call leg gets shown as placing a call into the conference bridge. The system shows the call as a call *into* the bridge, regardless of the actual direction of the call.

Field Names	Orig Call CDR	Setup Call CDR	Conference CDR 1	Conference CDR 2	Conference CDR 3	Final CDR
globalCallID_callId	1	2	1	1	1	1
origLegCallIdentifier	101	105	101	102	106	101
destLegCallIdentifier	102	106	115	116	117	102
callingPartyNumber	2001	2001	2001	2309	3071111	2001
originalCalledPartyNumber	2309	3071111	b0029901001	b0029901001	b0029901001	2309
finalCalledPartyNumber	2309	3071111	b0029901001	b0029901001	b0029901001	2309
lastRedirectDn	2001	3071111	b0029901001	b0029901001	b0029901001	b0029901001
origCause_Value	393216	0	16	393216	393216	16
dest_CauseValue	393216	0	393216	393216	393216	0
origCalledPartyRedirectReason	0	0	0	0	0	0
lastRedirectRedirectReason	0	0	0	0	0	98
origTerminationOnBehalfOf	4	4	12	12	4	12
destTerminationOnBehalfOf	4	4	0	0	4	4
origCalledRedirectOnBehalfOf	0	0	4	4	4	0
lastRedirectRedirectOnBehalfOf	0	0	4	4	4	4
joinOnBehalfOf	0	0	4	4	4	4
Conversation ID	0	1		1	1	0
duration	60	360		360	360	55

Comment	
Orig Call CDR	
Setup Call CDR	ConfControllerDn=2001;ConfControlerDeviceName=SEP0003E333FEBD
Conference CDR 1	ConfControllerDn=2001;ConfControlerDeviceName=SEP0003E333FEBD
Conference CDR 2	ConfControllerDn=2001;ConfControlerDeviceName=SEP0003E333FEBD

Conference CDR 3 Final CDR	ConfControllerDn=2001;ConfControlerDeviceName=SEP0003E333FEBD

Secure Meet-Me Conference

The following example shows a CDR for a meet-me secure conference. 35010 calls into a secure meet-me conference, but 35010 is a non-secure phone. Because 35010 does not meet the minimum security level of the meet-me conference, the call is cleared with the cause code of 58 (meet-me conference minimum security level not met).

Secure Conference Example

Field Names	Call to the Meet-Me Conference CDR
globalCallID_callId	5045247
origLegCallIdentifier	123456879
destLegCallIdentifier	123456999
callingPartyNumber	35010
originalCalledPartyNumber	50000
finalCalledPartyNumber	50000
lastRedirectDn	50000
origCause_Value	58
dest_CauseValue	0
origCalledPartyRedirectReason	0
lastRedirectRedirectReason	0
origCalledPartyRedirectOnBehalfOf	0
lastRedirectRedirectOnBehalfOf	0
origTerminationOnBehalfOf	6
destTerminationOnBehalfOf	6

Ad Hoc Conference Linking

The ad hoc conference linking feature generates many different CDRs depending on the circumstances of the conference. The following scenarios show some of the different CDRs:

- Conference Linking using Join, page 10-51
- Conference Linking Using Transfer or Direct Transfer, page 10-52
- Removing a Party From a Linked Conference, page 10-54
- Removing a Party (Controller) From a Linked Conference, page 10-56
- Removing the Linked Conference, page 10-58

Conference Linking using Join

The direction of the call between the bridges depends upon which of Carol's two calls is primary. The primary call survives and the secondary call is redirected to the conference.

Alice calls Bob, and Bob conferences in Carol (Conference 1). Dave calls Carol, and conferences in Ed (Conference 2). Two separate conferences are created. Carol is in both conferences. At this point CDR1, CDR2, CDR3, and CDR4 are generated.

Carol joins the two conferences. At this point CDR5 is generated.

When the remaining parties hang up, the remaining CDRs are generated in the order that the parties leave the conference.

Conference LInking using Join Example

Field Names	CDR1: Alice -> Bob (original call)	CDR2: Bob -> Carol (consultation call)	CDR3: Dave -> Carol (original call)	CDR4: Dave -> Ed (consultation call)	CDR5: Carol -> Conference Bridge (conference call)	CDR6: Dave -> Conference Bridge (conference call)
globalCallID_callId	1	2	3	4	3	3
origLegCallIdentifier	11	13	21	23	22	21
destLegCallIdentifier	12	14	22	24	25	26
callingPartyNumber	1000	1001	1003	1003	1002	1003
originalCalledPartyNumber	1001	1002	1002	1004	b0029901222	b0029901222
finalCalledPartyNumber	1001	1002	1002	1004	b0029901222	b0029901222
lastRedirectDn	1001	1002	1002	1004	1003	1003
origTerminationOnBehalfOf	4	4	4	4	4	0
destTerminationOnBehalfOf	4	4	4	4	4	0
lastRedirectRedirectReason	0	0	0	0	98	98
lastRedirectRedirectOnBehalfOf	0	0	0	0	4	4
origConversationID	0	0	0	0	0	0
destConversationID	0	0	0	0	2222	2222
Comment					ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1	ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1

Field Names	CDR7: Ed -> Conference Bridge (conference call)	CDR8 Dave -> Conference Bridge (conference call)	CDR9: Alice -> Conference Bridge (conference call)	CDR10: Bob -> Conference Bridge (conference call)	CDR11: Carol -> Conference Bridge (conference call)
globalCallID_callId	3	1	1	1	1
origLegCallIdentifier	24	25	11	12	14
destLegCallIdentifier	27	28	15	16	17
callingPartyNumber	1004	b0029901222	1000	1001	1002
originalCalledPartyNumber	b0029901222	b0029901001	b0029901001	b0029901001	b0029901001
finalCalledPartyNumber	b0029901222	b0029901001	b0029901001	b0029901001	b0029901001
lastRedirectDn	1003	1002	1001	1001	1001
origTerminationOnBehalfOf	0	0	0	0	0
destTerminationOnBehalfOf	0	0	0	0	0
lastRedirectRedirectReason	98	98	98	98	98
lastRedirectRedirectOnBehalfOf	4	4	4	4	4
origConversationID	0	2222			
destConversationID	2222	1111			
Comment	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`			

Conference Linking Using Transfer or Direct Transfer

Alice calls Bob, and Bob conferences Carol (Conference 1). Dave calls Carol, and conferences in Ed (Conference 2) Two separate conferences are created; Carol is in both conferences. At this point CDR1, CDR2, CDR3, and CDR4 are generated.

Carol presses the Direct Transfer (DirTrfr) softkey on the call to the first conference. Alice and Bob are in Conference 1 while Dave and Ed are in Conference 2. When the remaining parties hang up, the remaining CDRs are generated in the order the parties leave the conference.

Note The direction of the call between the bridges depends on which of Carol's two calls is the primary call. The primary call side is the calling party of the transferred call.

Conference Linking Using Transfer or Direct Transfer Example

Field Names	CDR1: Alice -> Bob (original call)	CDR2: Bob -> Carol (consultation call)	CDR3: Dave -> Carol (original call)	CDR4: Dave -> Carol (consultation call)	CDR5: Carol -> Conference Bridge (conference call)	CDR6: Carol -> Conference Bridge (conference call)
globalCallID_callId	1	2	3	4	1	3
origLegCallIdentifier	11	13	21	23	14	22
destLegCallIdentifier	12	14	22	24	17	25
callingPartyNumber	1000	1001	1003	1003	1002	1002
originalCalledPartyNumber	1001	1002	1002	1004	b0029901001	b0029901222
finalCalledPartyNumber	1001	1002	1002	1004	b0029901001	b0029901222
lastRedirectDn	1001	1002	1002	1004	1001	1003
origTerminationOnBehalfOf	4	4	4	4	10	10
destTerminationOnBehalfOf	4	4	4	4	10	10
lastRedirectRedirectReason	0	0	0	0	98	98
lastRedirectRedirectOnBehalfOf	0	0	0	0	4	4
origConversationID	0	0	0	0	0	0
destConversationID	0	0	0	0	1111	2222
Comment					`ConfControllerDn=1001;ConfControllerDeviceName=SEP0003E333FEBD;ConfRequestorDn-1001;ConfRequestorDeviceName=SEP0003E333FEBD`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`

Field Names	CDR7: Dave-> Conference Bridge (conference call)	CDR8: Ed -> Conference Bridge (conference call)	CDR9: Conference Bridge-> Conference Bridge	CDR-10: Alice -> Conference Bridge (conference call)	CDR11: Bob-> Conference Bridge (conference call)
globalCallID_callId	3	3	1	1	1
origLegCallIdentifier	21	24	17	11	12
destLegCallIdentifier	26	27	28	15	16
callingPartyNumber	1003	1004	b0029901001	1000	1001

originalCalledPartyNumber	b0029901222	b0029901222	b0029901222	b0029901001	b0029901001
finalCalledPartyNumber	b0029901222	b0029901222	b0029901222	b0029901001	b0029901001
lastRedirectDn	1003	1003	1002	1001	1001
origTerminationOnBehalfOf	0	0	0	0	0
destTerminationOnBehalfOf	0	0	0	0	0
lastRedirectRedirectReason	98	98	4	98	98
lastRedirectRedirectOnBehalfOf	4	4	10	4	4
origConversationID	0	0	1111	0	0
destConversationID	2222	2222	2222	1111	1111
Comment	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`	`ConfControllerDn=1001;ConfControllerDeviceName=SEP0003E333FEBD;ConfRequestorDn-1001;ConfRequestorDeviceName=SEP0003E333FEBD`	`ConfControllerDn=1001;ConfControllerDeviceName=SEP0003E333FEBD;ConfRequestorDn-1001;ConfRequestorDeviceName=SEP0003E333FEBD`

Removing a Party From a Linked Conference

CDRs are generated in the order the parties leave a conference. When the remaining conference only has two parties, the two parties are joined directly together.

Alice calls Bob, and Bob conferences Carol (Conference 1). Dave calls Carol, and conferences in Ed (Conference 2) Two separate conferences are created; Carol is in both conferences. At this point CDR1, CDR2, CDR3, and CDR4 are generated.

Carol presses the Direct Transfer (DirTrfr) softkey on the call to the first conference. Alice and Bob are in Conference 1 while Dave and Ed are in Conference 2. Conference 1 and Conference 2 are transferred together. Carol hangs up, and leaves only two parties in Conference 1.

Because there are only two parties in the conference, Bob and the conference link are joined together. At this point, CDR7, CDR8, and CDR9 are generated. Because Bob is the controller in Conference 1, Bob is the calling party in the call between Bob and Conference 2. When the remaining parties hang up, the remaining CDRs are generated in the order the parties leave the conference.

Note If Bob is not a controller and the chaining occurs before Bob joins Conference 1, the call between Bob and Conference 2 is generated in the opposite direction of what is shown in the CDRs.

The direction of the call between the final two parties of a conference depends on who has been in the conference the longest. The party that has been in the conference the longest becomes the calling party.

Removing a Party from Linked Conference Example

Field Names	CDR1: Alice -> Bob (original call)	CDR2: Bob -> Carol (consultation call)	CDR3: Dave -> Carol (original call)	CDR4: Dave -> Carol (consultation call)	CDR5: Carol -> Conference Bridge (conference call)	CDR6: Carol -> Conference Bridge (conference call)
globalCallID_callId	1	2	3	4	1	3
origLegCallIdentifier	11	13	21	23	14	22
destLegCallIdentifier	12	14	22	24	17	25
callingPartyNumber	1000	1001	1003	1003	1002	1002
originalCalledPartyNumber	1001	1002	1002	1004	b0029901001	b0029901222
finalCalledPartyNumber	1001	1002	1002	1004	b0029901001	b0029901222
lastRedirectDn	1001	1002	1002	1004	1001	1003
origTerminationOnBehalfOf	4	4	4	4	10	10
destTerminationOnBehalfOf	4	4	4	4	10	10
lastRedirectRedirectReason	0	0	0	0	98	98
lastRedirectRedirectOnBehalfOf	0	0	0	0	4	4
origConversationID	0	0	0	0	0	0
destConversationID	0	0	0	0	1111	2222
Comment					`ConfControllerDn=1001;ConfControllerDeviceName=SEP0003E333FEBD;ConfRequestorDn-1001;ConfRequestorDeviceName=SEP0003E333FEBD`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`

Field Names	CDR7: Alice-> Conference Bridge (conference call)	CDR8: Bob-> Conference Bridge (conference call)	CDR9: Conference Bridge-> Conference Bridge	CDR-10: Bob -> Conference Bridge (conference call)	CDR11: Dave-> Conference Bridge (conference call)	CDR12: Ed -> Conference Bridge (conference call)
globalCallID_callId	1	1	3	3	3	3
origLegCallIdentifier	11	12	25	11	12	24
destLegCallIdentifier	15	16	28	15	16	27

callingPartyNumber	1000	1001	b0029901222	1000	1001	1004
originalCalledPartyNumber	b0029901001	b0029901001	b0029901001	b0029901222	b0029901222	b0029901222
finalCalledPartyNumber	b0029901001	b0029901001	b0029901001	b0029901222	b0029901222	b0029901222
lastRedirectDn	1001	1001	1002	b0029901001	1003	1003
origTerminationOnBehalfOf	16	4	4	4	0	0
destTerminationOnBehalfOf	0	4	4	4	0	0
lastRedirectRedirectReason	98	98	4	98	98	98
lastRedirectRedirectOnBehalfOf	4	4	10	4	4	4
origConversationID	0	0	2222	0	0	0
destConversationID	1111	1111	1111	2222	2222	2222
Comment	`ConfControllerDn=1001;ConfControllerDeviceName=SEP0003E333FEBD;ConfRequestorDn-1001;ConfRequestorDeviceName=SEP0003E333FEBD`	`ConfControllerDn=1001;ConfControllerDeviceName=SEP0003E333FEBD;ConfRequestorDn-1001;ConfRequestorDeviceName=SEP0003E333FEBD`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`

Removing a Party (Controller) From a Linked Conference

CDRs are generated in the order the parties leave a conference. When the remaining conference only has two parties, these two parties are joined directly together.

Alice calls Bob, and Bob conferences Carol (Conference 1). Dave calls Carol, and conferences in Ed (Conference 2) Two separate conferences are created; Carol is in both conferences. At this point CDR1, CDR2, CDR3, and CDR4 are generated.

Carol presses the Direct Transfer (DirTrfr) softkey on the call to the first conference. Alice and Bob are in Conference 1, while Dave and Ed are in Conference 2. Conference 1 and Conference 2 are transferred together. Bob hangs up, leaving only two parties connected to Conference 1.

Because there are only two parties in Conference1, Alice and the conference link are joined directly together. At this point, CDR7, CDR8, and CDR9 are generated. Because Alice has been in the conference longer, she becomes the calling party in the call between Alice and Conference 2. When the remaining parties hang up, the remaining CDRs are generated in the order the parties leave the conference.

Note The direction of a call between the final two parties of a conference depends on who has been in the conference the longest. The party that has been in the conference the longest becomes the calling party.

Removing a Controller from a Linked Conference Example

Field Names	CDR1: Alice -> Bob (original call)	CDR2: Bob -> Carol (consultation call)	CDR3: Dave -> Carol (original call)	CDR4: Dave -> Carol (consultation call)	CDR5: Carol -> Conference Bridge (conference call)	CDR6: Carol -> Conference Bridge (conference call)
globalCallID_callId	1	2	3	4	1	3
origLegCallIdentifier	11	13	21	23	14	22
destLegCallIdentifier	12	14	22	24	17	25
callingPartyNumber	1000	1001	1003	1003	1002	1002
originalCalledPartyNumber	1001	1002	1002	1004	b0029901001	b0029901222
finalCalledPartyNumber	1001	1002	1002	1004	b0029901001	b0029901222
lastRedirectDn	1001	1002	1002	1004	1001	1003
origTerminationOnBehalfOf	4	4	4	4	10	10
destTerminationOnBehalfOf	4	4	4	4	10	10
lastRedirectRedirectReason	0	0	0	0	98	98
lastRedirectRedirectOnBehalfOf	0	0	0	0	4	4
origConversationID	0	0	0	0	0	0
destConversationID	0	0	0	0	1111	2222
Comment					`ConfControllerDn=1001;ConfControllerDeviceName=SEP0003E333FEBD;ConfRequestorDn-1001;ConfRequestorDeviceName=SEP0003E333FEBD`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`

Field Names	CDR7: Conference Bridge -> Conference Bridge	CDR8: Alice-> Conference Bridge (conference call)	CDR9: Conference Bridge-> Conference Bridge	CDR-10: Alice -> Conference Bridge (conference call)	CDR11: Dave-> Conference Bridge (conference call)	CDR12: Ed -> Conference Bridge (conference call)
globalCallID_callId	1	1	3	3	3	3
origLegCallIdentifier	12	11	25	11	21	24
destLegCallIdentifier	16	15	28	25	26	27

callingPartyNumber	1001	1000	b0029901222	1001	1003	1004
originalCalledPartyNumber	b0029901001	b0029901001	b0029901001	b0029901222	b0029901222	b0029901222
finalCalledPartyNumber	b0029901001	b0029901001	b0029901001	b0029901222	b0029901222	b0029901222
lastRedirectDn	1001	1001	1002	b0029901001	1003	1003
origTerminationOnBehalfOf	4	16	4	4	0	0
destTerminationOnBehalfOf	4	0	4	4	0	0
lastRedirectRedirectReason	98	98	4	98	98	98
lastRedirectRedirectOnBehalfOf	4	4	10	4	4	4
origConversationID	0	0	2222	0	0	0
destConversationID	1111	1111	1111	2222	2222	2222
Comment	ConfControllerDn=1001;ConfControllerDeviceName=SEP0003E333FEBD;ConfRequestorDn-1001;ConfRequestorDeviceName=SEP0003E333FEBD	ConfControllerDn=1001;ConfControllerDeviceName=SEP0003E333FEBD;ConfRequestorDn-1001;ConfRequestorDeviceName=SEP0003E333FEBD	ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1	ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1	ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1	ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1

Removing the Linked Conference

Alice calls Bob, and Bob conferences Carol (Conference 1). Dave calls Carol, and conferences in Ed (Conference 2) Two separate conferences are created; Carol is in both conferences. At this point CDR1, CDR2, CDR3, and CDR4 are generated.

Carol presses the **Direct Transfer** (DirTrfr) softkey on the call to the first conference. Alice and Bob are in Conference 1, while Dave and Ed are in Conference 2. Conference 1 and Conference 2 are transferred together.

Bob presses the ConfList softkey and has Alice, Bob, and the conference link “Conference” shown in the list. Bob selects “Conference” and presses the **Remove** softkey. At this point, CDR7, CDR8, and CDR9 are generated. The conference link is removed, leaving two parties in the conference.

The remaining two parties are joined together. In Conference 1, Alice and Bob are joined together, and in Conference 2, Dave and Ed are joined together. When the remaining parties hang up, the remaining CDRs are generated in the order the parties leave the conference.

Removing the Linked Conference Example

Field Names	CDR1: Alice -> Bob (original call)	CDR2: Bob -> Carol (consultation call)	CDR3: Dave -> Carol (original call)	CDR4: Dave -> Carol (consultation call)	CDR5: Carol -> Conference Bridge (conference call)	CDR6: Carol -> Conference Bridge (conference call)
globalCallID_callId	1	2	3	4	1	3
origLegCallIdentifier	11	13	21	23	14	22
destLegCallIdentifier	12	14	22	24	17	25
callingPartyNumber	1000	1001	1003	1003	1002	1002
originalCalledPartyNumber	1001	1002	1002	1004	b0029901001	b0029901222
finalCalledPartyNumber	1001	1002	1002	1004	b0029901001	b0029901222
lastRedirectDn	1001	1002	1002	1004	1001	1003
origTerminationOnBehalfOf	4	4	4	4	10	10
destTerminationOnBehalfOf	4	4	4	4	10	10
lastRedirectRedirectReason	0	0	0	0	98	98
lastRedirectRedirectOnBehalfOf	0	0	0	0	4	4
origConversationID	0	0	0	0	0	0
destConversationID	0	0	0	0	1111	2222
Comment					`ConfControllerDn=1001;ConfControllerDeviceName=SEP0003E333FEBD;ConfRequestorDn-1001;ConfRequestorDeviceName=SEP0003E333FEBD`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`

Field Names	CDR7: Conference Bridge > Conference Bridge	CDR8: Alice-> Conference Bridge (conference call)	CDR9: Bob -> Conference Bridge	CDR-10: Dave-> Conference Bridge (conference call)	CDR11: Ed-> Conference Bridge (conference call)	CDR12: Bob -> Alice
globalCallID_callId	3	1	1	3	3	3
origLegCallIdentifier	25	11	12	21	24	21
destLegCallIdentifier	28	15	16	26	27	24

callingPartyNumber	b0029901222	1000	1001	1003	1004	1003
originalCalledPartyNumber	b0029901001	b0029901001	b0029901001	b0029901222	b0029901222	b0029901222
finalCalledPartyNumber	b0029901001	b0029901001	b0029901001	b0029901222	b0029901222	1004
lastRedirectDn	1002	1001	1001	1003	1003	b0029901222
origTerminationOnBehalfOf	4	4	4	16	0	0
destTerminationOnBehalfOf	4	4	4	0	0	0
lastRedirectRedirectReason	4	98	98	98	98	98
lastRedirectRedirectOnBehalfOf	10	4	4	4	4	4
origConversationID	2222	0	0	0	0	0
destConversationID	1111	1111	1111	2222	2222	0
Comment	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`	`ConfControllerDn=1001;ConfControllerDeviceName=SEP0003E333FEBD;ConfRequestorDn-1001;ConfRequestorDeviceName=SEP0003E333FEBD`	`ConfControllerDn=1001;ConfControllerDeviceName=SEP0003E333FEBD;ConfRequestorDn-1001;ConfRequestorDeviceName=SEP0003E333FEBD`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`

Field Names	CDR13: Dave -> Ed
globalCallID_callId	3
origLegCallIdentifier	21
destLegCallIdentifier	24
callingPartyNumber	1003
originalCalledPartyNumber	b0029901222
finalCalledPartyNumber	1004
lastRedirectDn	b0029901222
origTerminationOnBehalfOf	0
destTerminationOnBehalfOf	0
lastRedirectRedirectReason	98
lastRedirectRedirectOnBehalfOf	4
origConversationID	0

destConversationID	0
Comment	`ConfControllerDn=1003;ConfControllerDeviceName=SEP0003E333FAD1;ConfRequestorDn-1003;ConfRequestorDeviceName=SEP0003E333FAD1`

Call Park

Call Park will generate two CDRs, one for the original call that is parked and another for the call that is picked up or reverted. These CDRs will have the same globalCallID_callId. This section contains the following CDR examples:

- Call Park Pickup, page 10-61
- Call Park Reversion, page 10-62

Call Park Pickup

When the call is parked, the call gets split. The original call generates a CDR. The **origTerminationOnBehalfOf** and **destTerminationOnBehalfOf** fields get set to (Call Park = 3) for this CDR.

When the parked call is retrieved, the user goes off hook and enters the park code. This call joins with the parked call. Because the user picking up is joined with the parked call, the system treats the user as the originator of the call, and the parked user gets treated as the destination. This means that the **callingPartyNumber** of the call contains the directory number of the user who is picking up the call and the **originalCalledNumber** and **finalCalledNumber** contain the directory number of the parked user. The **lastRedirectDn** contains the park code that is used to pick up the call. The **lastRedirectRedirectReason** specifies (Call Park Pickup = 8). The **lastRedirectRedirectOnBehalfOf** should specify (Call Park = 3).

Call Park Example

50003 calls 50002; 50002 presses the Park softkey. 50001 picks up the parked call by dialing the park code (44444).

Field Names	Original Call That Is Parked CDR	Parked Call That Is Picked Up CDR
globalCallID_callId	1	1
origLegCallIdentifier	20863957	20863961
destLegCallIdentifier	20863958	20863957
callingPartyNumber	50003	50001
originalCalledPartyNumber	50002	50003

finalCalledPartyNumber	50002	50003
lastRedirectDn	50002	44444
origCause_Value	393216	0
dest_CauseValue	393216	16
origCalledPartyRedirectReason	0	0
lastRedirectRedirectReason	0	8
origCalledPartyRedirectOnBehalfOf	0	0
lastRedirectRedirectOnBehalfOf	0	3
origTerminationOnBehalfOf	3	0
destTerminationOnBehalfOf	3	12
joinOnBehalfOf	0	3
duration	4	60

Call Park Reversion

When a call is parked and not picked up, the call park reversion timer will expire and redirect the call back to the called party. In this case, Two CDRs get generated. The first CDR acts the same as the Call Park Pickup scenario, but the second CDR slightly differs. When the Call Pickup Reversion timer expires, the call redirects to the called party.

When the call is parked, the call gets split. This action generates a CDR for the original call. The **origTerminationOnBehalfOf** and **destTerminationOnBehalfOf** fields get set to (Call Park = 3) for this CDR (same as Call Park Pickup scenario).

When the Call Park Reversion timer expires, the call gets redirected to the called party. The **origCalledPartyRedirectOnBehalfOf** and **lastRedirectRedirectOnBehalfOf** fields specify Call Park = 3. The **origCalledPartyRedirectReason** field specifies Call Park = 7, and the **lastRedirectRedirectReason** field specifies Call Park Reversion = 11.

Call Park Reversion Example

- **Call Park Reversion Example** – 50003 calls 50002, 50002 presses the Park softkey. Nobody picks up the parked call; it reverts to 50002, and 50002 answers.

Field Names	Original Call That Is Parked CDR	Reverted Call CDR
globalCallID_callId	2	2
origLegCallIdentifier	20863963	20863963
destLegCallIdentifier	20863964	20863967
callingPartyNumber	50003	50003
originalCalledPartyNumber	50002	50002
finalCalledPartyNumber	50002	50002
lastRedirectDn	50002	50002
origCause_Value	393216	0
dest_CauseValue	393216	16

origCalledPartyRedirectReason	0	7
lastRedirectRedirectReason	0	11
origCalledPartyRedirectOnBehalfOf	0	3
lastRedirectRedirectOnBehalfOf	0	3
origTerminationOnBehalfOf	3	3
destTerminationOnBehalfOf	3	12
joinOnBehalfOf	0	3
duration	7	60

Precedence Calls (MLPP)

With precedence calls, everything basically remains the same for all calls (normal calls, forwarded calls, transferred calls, and so forth). The difference occurs because the precedence level fields are set in the CDR. Also, when a call is preempted by a higher level precedence call, the cause codes indicate the reason for the preemption.

Precedence Call Examples

- Call to another IP phone by dialing a precedence pattern (precedence level 2).

Field Names	Precedence Call CDR
globalCallID_callId	100
origLegCallIdentifier	12345
destLegCallIdentifier	12346
callingPartyNumber	2001
origCalledPartyNumber	826001
origCause_Value	0
dest_CauseValue	16
origPrecedenceLevel	2
destPrecedenceLevel	2

- Received precedence call from another network (precedence level 1)

Field Names	Precedence Call CDR
globalCallID_callId	102
origLegCallIdentifier	11111
destLegCallIdentifier	11112
callingPartyNumber	9728552001
origCalledPartyNumber	6001
origCause_Value	16

dest_CauseValue	0
origPrecedenceLevel	1
destPrecedenceLevel	1

- Call gets preempted by a higher precedence level call.

Field Names	Original call CDR	Higher Level Call CDR
globalCallID_callId	10000	10001
origLegCallIdentifier	12345678	12345680
destLegCallIdentifier	12345679	12345681
callingPartyNumber	2001	9728551234
origCalledPartyNumber	826001	826001
origCause_Value	0	0
dest_CauseValue	9	16
origPrecedenceLevel	2	1
destPrecedenceLevel	2	1

Malicious Calls

When a call is identified as a malicious call (button press), the local network (CCM) flags the call. The “comment” field gets used to flag the malicious call.

Malicious Call Example

- Customer call marked as malicious.

Field Names	Original call CDR
globalCallID_callId	1
origLegCallIdentifier	100
destLegCallIdentifier	101
callingPartyNumber	9728552001
origCalledPartyNumber	5555
origCause_Value	0
dest_CauseValue	16
Comment	callFlag=MALICIOUS

Immediate Divert (to Voicemail)

Immediate Divert (IDivert) can get invoked in three different call states:

- The IDivert feature can get invoked while the incoming call is ringing. The CDR for the ringing case acts very similar to call forwarding, but the **origCalledPartyRedirectOnBehalfOf** and the **lastRedirectRedirectOnBehalfOf** specifies (Immediate Divert = 14).
- The IDivert feature can get invoked while the call is connected or on hold. These scenarios generate two CDRs. Both CDRs will have the same **globalCallID_CallId** field. The first CDR applies to the original connection, and the second CDR applies to the call redirected to voice-messaging system. The first call will have the **origTerminationOnBehalfOf** and **destTerminationOnBehalfOf** field set to (Immediate Divert = 14).
- The call that is redirected to the voice-messaging system will have the **origCalledPartyRedirectOnBehalfOf** and the **lastRedirectRedirectOnBehalfOf** set to (Immediate Divert = 14).

IDivert Examples

- **IDivert during Alerting –** 40003 calls 40001 and while 40001 is ringing, 40001 presses the IDivert button and call diverts to the voice-messaging system (40000).

Note If the call is redirected by IDivert in the Alerting state, only one CDR is generated.

Field Names	Original call CDR
globalCallID_callId	37
origLegCallIdentifier	16777327
destLegCallIdentifier	16777329
callingPartyNumber	40003
origCalledPartyNumber	40001
finalCalledPartyNumber	40000
lastRedirectDn	40001
origCause_Value	16
dest_CauseValue	0
origCalledPartyRedirectReason	50
lastRedirectRedirectReason	50
origCalledPartyRedirectOnBehalfOf	14
lastRedirectRedirectOnBehalfOf	14
joinOnBehalfOf	14

- **IDivert during Connect –** 40003 calls 40001, and 40001 answers the call. 40001 decides to divert the caller to the voice-messaging system and presses the IDivert softkey. 40003 gets diverted to the voice-messaging system (40000).

  Because the call was connected before the redirect, two CDRs get generated: one for the original connected call and another for the call diverted to the voice-messaging system.

Field Names	Original Connected Call CDR	Diverted Call CDR
globalCallID_callId	38	38

origLegCallIdentifier	16777330	16777330
destLegCallIdentifier	16777331	16777332
callingPartyNumber	40003	40003
origCalledPartyNumber	40001	40001
finalCalledPartyNumber	40001	40000
lastRedirectDn	40001	40001
origCause_Value	0	16
dest_CauseValue	0	0
origCalledPartyRedirectReason	0	50
lastRedirectRedirectReason	0	50
origCalledPartyRedirectOnBehalfOf		14
lastRedirectRedirectOnBehalfOf		14
origTerminationOnBehalfOf	14	14
destTerminationOnBehalfOf	14	12
joinOnBehalfOf		14

Barge

When a shared line uses the barge feature, the **origCalledPartyNumber**, **finalCalledPartyNumber** and **lastRedirectDn** represent the conference bridge number ‘b00…’. The redirect and join OnBehalfOf fields have a value of (Barge = 15), and the redirect reason fields specify (Barge = 114).

Barge Examples

- **Barge Example 1–** 40003 calls 40001 and 40001 answers. Shared line 40001’ on another phone presses the Barge softkey. All the parties get conferenced together; then, 40003 hangs up.

Note Both CDRs have the same globalCallID_callId, and the conversationID field links back to the CI (call Identifier) of the barged call.

Field Names	Original Call CDR	Barge Call CDR
globalCallID_callId	7	7
origLegCallIdentifier	16777230	16777232
destLegCallIdentifier	16777231	16777235
callingPartyNumber	40003	40003
origCalledPartyNumber	40001	b001501001
finalCalledPartyNumber	40001	b001501001
lastRedirectDn	40001	b001501001
origCause_Value	16	0
dest_CauseValue	0	0
origCalledPartyRedirectReason	0	114

lastRedirectRedirectReason	0	114
origCalledPartyRedirectOnBehalfOf		15
lastRedirectRedirectOnBehalfOf		15
joinOnBehalfOf		15
destConversationID	0	16777231

- **Barge Example 2–** 40003 calls 40001, and 40001 answers. Shared line 40001’ on another phone presses the Barge softkey. All the parties get conferenced together; then, 40001 hangs up.

Note Both CDRs have the same globalCallID_callId, and the conversationID field links back to the CI (call Identifier) of the barged call.

Field Names	Original Call CDR	Barge Call CDR	Final Call CDR
globalCallID_callId	9	9	9
origLegCallIdentifier	16777236	16777238	16777236
destLegCallIdentifier	16777237	16777241	16777238
callingPartyNumber	40003	40001	40003
origCalledPartyNumber	40001	b001501001	40001
finalCalledPartyNumber	40001	b001501001	40001
lastRedirectDn	40001	b001501001	40001
origCause_Value	0	393216	16
dest_CauseValue	16	393216	0
origCalledPartyRedirectReason	0	114	0
lastRedirectRedirectReason	0	114	0
origTerminationOnBehalfOf		15	12
destTerminationOnBehalfOf	12	15	12
lastRedirectRedirectOnBehalfOf		15	
joinOnBehalfOf		15	
destConversationID	0	16777237	0

- **Barge Example 3–** 40003 calls 40001, and 40001 answers. Shared line 40001’ on another phone presses the Barge softkey. All the parties get conferenced together; then, 40001’ (another shared line and phone) presses the Barge softkey. 40003 hangs up first.

Note All CDRs have the same **globalCallID_callId,** and the conversationID field links back to the CI (call Identifier) of the barged call.

Field Names	Original Call CDR	Barge Call 1 CDR	Barge Call 2 CDR
globalCallID_callId	14	14	14

origLegCallIdentifier	16777249	16777251	16777255
destLegCallIdentifier	16777250	16777254	16777258
callingPartyNumber	40003	40001	40001
origCalledPartyNumber	40001	b001501001	b001501001
finalCalledPartyNumber	40001	b001501001	b001501001
lastRedirectDn	40001	b001501001	b001501001
origCause_Value	16	0	0
dest_CauseValue	0	0	0
origCalledPartyRedirectReason	0	114	114
lastRedirectRedirectReason	0	114	114
origTerminationOnBehalfOf	12	15	15
destTerminationOnBehalfOf			
origRedirectOnBehalfOf		15	15
lastRedirectRedirectOnBehalfOf		15	15
joinOnBehalfOf		15	15
destConversationID	0	16777250	16777251

cBarge

The cBarge feature acts very similar to the conference feature. When a shared line uses the cBarge feature, the **origCalledPartyNumber, finalCalledPartyNumber** and **lastRedirectDn** represent the conference bridge number ‘b00…’. The redirect and join **OnBehalfOf** fields have a value of (Conference = 4), and the **redirect reason** fields specify (Conference = 98).

cBarge Examples

- **cBarge Example –** 40003 calls 40001, and 40001 answers; 40001’ (shared line) on another phone presses the cBarge button.

Field Names	Orig Call CDR	cBarge Call CDR 1	cBarge Call CDR 2	cBarge Call CDR 3	Final Call CDR
globalCallID_callId	49	49	49	49	49
origLegCallIdentifier	1677346	1677348	1677347	1677346	1677347
destLegCallIdentifier	1677347	1677353	1677351	1677352	1677346
callingPartyNumber	40003	40001	40001	40003	40001
originalCalledPartyNumber	40001	b0029901001	b0029901001	b0029901001	40003
finalCalledPartyNumber	40001	b0029901001	b0029901001	b0029901001	40003
lastRedirectDn	40001	b0029901001	40001	40001	b0029901001
origCause_Value	393216	16	393216	393216	16
dest_CauseValue	393216	0	393216	393216	0
origCalledPartyRedirectReason	0	98	98	98	0
lastRedirectRedirectReason	0	98	98	98	98

destTerminationOnBehalfOf	4		4	4	4
origCalledRedirectOnBehalfOf		4	4	4	
lastRedirectRedirectOnBehalfOf		4	4	4	4
joinOnBehalfOf		4	4	4	4
Conversation ID	0	16777220	16777220	16777220	1
duration	60	360		360	360

Comment	
Orig Call CDR	
cBarge Call CDR 1	ConfControllerDn=40003;ConfControlerDeviceName=SEP0003E333FEBD
cBarge Call CDR 2	ConfControllerDn=40003;ConfControlerDeviceName=SEP0003E333FEBD
cBarge Call CDR 3	ConfControllerDn=40003;ConfControlerDeviceName=SEP0003E333FEBD
Final Call CDR	ConfControllerDn=40003;ConfControlerDeviceName=SEP0003E333FEBD

Video Calls

The following example shows a CDR for a video call.

Video Calls Example

- **Example** - Calling party 51234 calls the called party 57890. In the following example, let 100 = H.261, 187962284 = 172.19.52.11, 288625580 = 172.19.52.17, 320 = 320K, and 2 = QCIF.

Field Names	Video Call CDR
globalCallID_callId	121
origLegCallIdentifier	101
destLegCallIdentifier	102
callingPartyNumber	51234
origCalledPartyNumber	57890
finalCalledPartyNumber	57890
lastRedirectDn	57890
origCause_Value	0
dest_CauseValue	16
origVideoCap_Codec	100
origVideoCap_Bandwidth	320
origVideoCap_Resolution	2
origVideoTransportAddress_IP	187962284
origVideoTransportAddress_Port	49208
destVideoCap_Codec	100
destVideoCap_Bandwidth	320

destVideoCap_Resolution	2
destVideoTransportAddress_IP	288625580
destVideoTransportAddress_Port	49254

Forced Authorization Code (FAC)

When the FAC feature is invoked, the authorization description, authorization level, and authorization code value get written into the CDR.

- The **authCodeDescription** field contains the description of the authorization code.
- The **authorizationLevel** field contains the level of authorization that is associated with the authorization code.
- The **authorizationCodeValue** field contains the authorization code.

Note The **authorizationCodeValue** field only displays in the CDR when the Display FAC in CDR service parameter is set to True. The default value of the parameter is False. See the "Configuring CDR Service Parameters" section on page 2-2.

FAC Example

45000 calls 9728134987; the system prompts the user for an authorization code and enters 12345. FAC code 12345 gets configured as level 1 and name Legal1. The caller answers the call and talks for 2 minutes. The Display FAC in CDR service parameter is set to True.

Field Names	Values
globalCallID_callId	100
origLegCallIdentifier	16777123
destLegCallIdentifier	16777124
callingPartyNumber	45000
origCalledPartyNumber	9728134987
finalCalledPartyNumber	9728134987
lastRedirectDn	9728134987
origCause_Value	0
dest_CauseValue	16
authCodeDescription	Legal1
authorizationLevel	1
duration	120
authorizationCodeValue	12345

Client Matter Code (CMC)

When the CMC feature is invoked, the client matter code gets written into the CDR. The **clientMatterCode** field contains the client matter code that the caller entered.

CMC Example

- 10000 calls 2142364624; the user gets prompted for a client matter code and enters 11111. The caller answers the call and talks for 10 minutes.

Field Names	Values
globalCallID_callId	101
origLegCallIdentifier	16777130
destLegCallIdentifier	16777131
callingPartyNumber	10000
origCalledPartyNumber	2142364624
finalCalledPartyNumber	2142364624
lastRedirectDn	2142364624
origCause_Value	0
dest_CauseValue	16
clientMatterCode	11111
duration	600

Call Secured Status

This field identifies security status of the call. It contains the highest level of security that is reached during a call. For example, if the call is originally unsecured, then later the call changed to secured, the CDR contains 1 for “Secured” even though different portions of the call had different status values. The **callSecuredStatus** field will identify the security status of the call.

Call Secured Status Examples

- **Encrypted Call Example** - The system encrypts the call between 20000 and 20001. The parties talk for 5 minutes.

Field Names	CDR
globalCallID_callId	102
origLegCallIdentifier	16777140
destLegCallIdentifier	16777141
callingPartyNumber	20000
origCalledPartyNumber	20001
finalCalledPartyNumber	20001
lastRedirectDn	20001
origCause_Value	0
dest_CauseValue	16
callSecuredStatus	2
duration	300

- **Authenticated Call Example** - The call between 20000 and 20001 gets authenticated (not encrypted). They talk for 10 minutes.

Field Names	CDR
globalCallID_callId	103
origLegCallIdentifier	16777142
destLegCallIdentifier	16777143
callingPartyNumber	20000
origCalledPartyNumber	20001
finalCalledPartyNumber	20001
lastRedirectDn	20001
origCause_Value	0
dest_CauseValue	16
callSecuredStatus	1
duration	600

DTMF Method

These fields identify the DTMF method that is used for the call.

DTMF Call Examples

- **No Preference Example** - The DTMF method that is used during this call represents No Preference/Best Effort. This call stays connected for 1 minute.

Field Names	CDR
globalCallID_callId	200
origLegCallIdentifier	16777500
destLegCallIdentifier	16777501
callingPartyNumber	20000
origCalledPartyNumber	20001
finalCalledPartyNumber	20001
lastRedirectDn	20001
origCause_Value	0
dest_CauseValue	16
origDTMFMethod	0
destDTMFMethod	0
duration	60

- **Preferred OOB Example** - The DTMF method that is used during this call represents OOB Preferred. This call remains connected for 1 minute.

Field Names	CDR
globalCallID_callId	201
origLegCallIdentifier	16777502
destLegCallIdentifier	16777503
callingPartyNumber	20000
origCalledPartyNumber	20001
finalCalledPartyNumber	20001
lastRedirectDn	20001
origCause_Value	0
dest_CauseValue	16
origDTMFMethod	1
destDTMFMethod	1
duration	60

RSVP

These fields identify the status of RSVP reservation for the call. The Cisco Unified Communications Manager RSVP CDR status field value gets concatenated, and the system retains the last 32 status values for the call.

For example, if a call is established with "Optional" policy, and the initial RSVP reservation is successful, and then it subsequently loses its bandwidth reservation and then regains its bandwidth reservation after retry, for several times during middle of the call, the call ends with a successful RSVP reservation. The CDR shows the following string as the Unified Communication RSVP reservation status for that particular stream: "2:5:2:5:2:5:2" (success:lost_bw:success:lost_bw:success:lost_bw:success).

RSVP Call Examples

- A call gets established with "Optional" policy, and the initial RSVP reservation succeeds. The parties talk for 5 minutes.

Field Names	CDR
globalCallID_callId	300
origLegCallIdentifier	16777300
destLegCallIdentifier	16777301
callingPartyNumber	20000
origCalledPartyNumber	20001
finalCalledPartyNumber	20001
lastRedirectDn	20001
origCause_Value	0
dest_CauseValue	16
origDTMFMethod	2

destDTMFMethod	2
duration	300

- The system establishes a call with "Optional" policy, and the initial RSVP reservation succeeds, then it loses its bandwidth reservation but regains it after a retry. The parties talk for 1 minute.

Field Names	CDR
globalCallID_callId	301
origLegCallIdentifier	16777302
destLegCallIdentifier	16777303
callingPartyNumber	20000
origCalledPartyNumber	20001
finalCalledPartyNumber	20001
lastRedirectDn	20001
origCause_Value	0
dest_CauseValue	16
origDTMFMethod	2:5:2
destDTMFMethod	2:5:2
duration	60

Redirection (3xx) Calls

The following example CDRs apply for a the redirection feature (3xx).

When a call is redirected by the Redirection Feature (3xx), the **origCalledPartyRedirectOnBehalfOf** and **lastRedirectRedirectOnBehalfOf** fields specify (Unified CM Redirection = 19). The **origCalledPartyRedirectReason** and the **lastRedirectRedirectReason** specify (Redirection = 162).

Redirection (3xx) Examples

- **Redirection Example** – Activate CFA on SIP phone 10010 (registered to Cisco Unified Communications Manager) with a CFA destination of 10000. 35010 calls 10010 which is CFA to 10000. The call gets redirected from 10010 to 10000. 10000 answers the call and talks for 1 minute.

Field Names	Original Call CDR
globalCallID_callId	11
origLegCallIdentifier	21832023
destLegCallIdentifier	21832026
callingPartyNumber	35010
originalCalledPartyNumber	10010
finalCalledPartyNumber	10000
lastRedirectDn	10010
origCause_Value	0
dest_CauseValue	16

origCalledPartyRedirectReason	162
lastRedirectRedirectReason	162
origCalledPartyRedirectOnBehalfOf	19
lastRedirectRedirectOnBehalfOf	19
origTerminationOnBehalfOf	0
destTerminationOnBehalfOf	12
joinOnBehalfOf	19
duration	60

Replaces Calls

The following examples display CDRs for various types of Replaces calls.

Replaces Examples

- **Invite with Replaces** – SIP phone 35010 calls SIP phone 35020. The transfer button gets pressed on 35010, and a call is placed to SCCP phone 3000, 3000 answers the call; then, SIP phone 35010 completes the transfer. The final transferred call occurs between 35020 and 3000.

Note When the transfer is complete, an Invite with Replaces is sent to Cisco Unified Communications Manager.

Field Names	Original Call CDR	Reverted Call CDR
globalCallID_callId	5045247	5045248
origLegCallIdentifier	21822467	21822469
destLegCallIdentifier	21822468	21822468
callingPartyNumber	35010	35020
originalCalledPartyNumber	3000	3000
finalCalledPartyNumber	3000	3000
lastRedirectDn	3000	35010
origCause_Value	393216	0
dest_CauseValue	393216	16
origCalledPartyRedirectReason	0	0
lastRedirectRedirectReason	0	146
origCalledPartyRedirectOnBehalfOf	0	0
lastRedirectRedirectOnBehalfOf	0	18
origTerminationOnBehalfOf	18	0
destTerminationOnBehalfOf	18	12
joinOnBehalfOf	0	18
duration	5	60

- **Refer with Replaces** – SIP phone 35010 calls SCCP 3000, the transfer button gets pressed on 35010, and a call is placed to SCCP 3001, 3001 answers the call; then, the SIP phone 35010 completes the transfer. The final transferred call occurs between 3000 and 3001.

Note When the transfer completes, a Refer with Replaces gets sent to Cisco Unified Communications Manager.

Field Names	Original Call CDR	Consultation Call CDR	Final Transferred Call CDR
globalCallID_callId	5045245	5045246	5045245
origLegCallIdentifier	21822461	21822463	21822462
destLegCallIdentifier	21822462	21822464	21822464
callingPartyNumber	35010	35010	3000
originalCalledPartyNumber	3000	3001	3001
finalCalledPartyNumber	3000	3001	3001
lastRedirectDn	3000	3001	35010
origCause_Value	393216	393216	16
dest_CauseValue	393216	393216	0
origCalledPartyRedirectReason	0	0	130
lastRedirectRedirectReason	0	0	146
origCalledPartyRedirectOnBehalfOf	0	0	17
lastRedirectRedirectOnBehalfOf	0	0	18
origTerminationOnBehalfOf	17	18	12
destTerminationOnBehalfOf	17	18	17
joinOnBehalfOf	0	0	18
duration	25	4	25

Refer Calls

See the "Replaces Calls" section on page 10-75 for an example of Refer with Replaces.

Monitor Calls

The following examples show CDRs for Monitor calls.

Monitor Examples

- **Monitor Example 1** – The customer (9728134987) calls the agent (30000), and the agent answers. The supervisor (40003) monitors the call. The **destConversationID** from the monitoring call matches the **destLegCallIdentifier** of the monitored call.

Field Names	Monitored Call CDR	Monitoring Call CDR
globalCallID_callId	7	10
origLegCallIdentifier	16777230	16777232
destLegCallIdentifier	16777231	16777235
callingPartyNumber	9728134987	40003
originalCalledPartyNumber	30000	b001501001
finalCalledPartyNumber	30000	b001501001
lastRedirectDn	30000	b001501001
origCause_Value	16	0
dest_CauseValue	0	0
origCalledPartyRedirectReason	0	370
lastRedirectRedirectReason	0	370
origCalledPartyRedirectOnBehalfOf		28
lastRedirectRedirectOnBehalfOf		28
destConversationID	0	16777231

- **Monitor Example 2** – The agent (30000) calls the customer (9728134987), and the customer answers. The supervisor (40003) monitors the call. The **destConversationID** from the monitoring call matches the **origLegCallIdentifier** of the monitored call.

Field Names	Monitored Call CDR	Monitoring Call CDR
globalCallID_callId	71	101
origLegCallIdentifier	16777299	16777932
destLegCallIdentifier	16777300	16777935
callingPartyNumber	30000	40003
originalCalledPartyNumber	9728134987	b001501002
finalCalledPartyNumber	9728134987	b001501002
lastRedirectDn	9728134987	b001501002
origCause_Value	16	0
dest_CauseValue	0	0
origCalledPartyRedirectReason	0	370
lastRedirectRedirectReason	0	370
origCalledPartyRedirectOnBehalfOf		28
lastRedirectRedirectOnBehalfOf		28
destConversationID	0	16777299

Recording Calls

The following examples show CDRs for Recording Calls.

Recording Calls Examples

- **Recording Calls Example 1** – The customer (9728134987) calls the agent (30000), and the agent answers. The recording feature creates two recording calls to the recording device and results in two additional CDRs: one for the agent voice and another for the customer voice. The **origConversationID** from the recording CDRs match the **destLegCallIdentifier** of the recorded call. In this example, the customer hangs up.

Field Names	Recorded Call CDR	Recording Call CDR1	Recording Call CDR2
globalCallID_callId	7	10	11
origLegCallIdentifier	16777110	16777120	16777122
destLegCallIdentifier	16777111	16777121	16777123
callingPartyNumber	9728134987	30000	30000
originalCalledPartyNumber	30000	90000	90000
finalCalledPartyNumber	30000	90000	90000
lastRedirectDn	30000	90000	90000
origCause_Value	16	0	0
dest_CauseValue	0	0	0
origCalledPartyRedirectReason	0	354	354
lastRedirectRedirectReason	0	354	354
origCalledPartyRedirectOnBehalfOf		27	27
lastRedirectRedirectOnBehalfOf		27	27
destConversationID	0	16777111	16777111

- **Recording Calls Example 2** – The agent (30000) calls the customer (9728134987), and the customer answers. The recording feature creates two recording calls to the recording device and results in two additional CDRs: one for the agent voice and another for the customer voice. The **origConversationID** from the recording CDRs matches the **origLegCallIdentifier** of the recorded call. In this example, the agent hangs up.

Field Names	Recorded Call CDR	Recording Call CDR1	Recording Call CDR2
globalCallID_callId	71	100	110
origLegCallIdentifier	16777113	16777220	16777222
destLegCallIdentifier	16777114	16777221	16777223
callingPartyNumber	30000	30000	30000
originalCalledPartyNumber	9728134987	90000	90000
finalCalledPartyNumber	9728134987	90000	90000
lastRedirectDn	9728134987	90000	90000
origCause_Value	16	16	16
dest_CauseValue	0	0	0
origCalledPartyRedirectReason	0	354	354

lastRedirectRedirectReason	0	354	354
origCalledPartyRedirectOnBehalfOf		27	27
lastRedirectRedirectOnBehalfOf		27	27
destConversationID	0	16777113	16777113

AAC and iLBC Calls

The following examples show CDRs for AAC and iLBC calls.

AAC Call Example

- This example applies to a call with AAC codec.

Field Names	AAC CDR
globalCallID_callId	121
origLegCallIdentifier	101
destLegCallIdentifier	102
callingPartyNumber	51234
originalCalledPartyNumber	57890
finalCalledPartyNumber	57890
lastRedirectDn	57890
origCause_Value	0
dest_CauseValue	16
origMediaCap_payloadCapability	42
origMediaCap_Bandwidth	256
destMediaCap_payloadCapability	42
destMediaCap_Bandwidth	256

iLBC Call Example

- This example applies to a call with iLBC codec.

Field Names	iLBC CDR
globalCallID_callId	121
origLegCallIdentifier	101
destLegCallIdentifier	102
callingPartyNumber	51234
originalCalledPartyNumber	57890
finalCalledPartyNumber	57890
lastRedirectDn	57890
origCause_Value	0

dest_CauseValue	16
origMediaCap_payloadCapability	86
origMediaCap_Bandwidth	15
destMediaCap_payloadCapability	86
destMediaCap_Bandwidth	15

Mobility

The following examples show CDRs for mobility calls.

Mobility Examples

- **Mobility Follow Me Example** - A dual-mode phone with the Enterprise number of 22285 and the cell number of 9728324124. 22202 calls 22285, and both 22285 and 9728324124 ring. The cell phone answers the call. The system generates a single CDR for this Follow Me call. The parties talk for 80 seconds.

Field Names	Follow Me Call CDR
globalCallID_callId	861
origLegCallIdentifier	22481077
destLegCallIdentifier	22481078
callingPartyNumber	22202
originalCalledPartyNumber	22285
finalCalledPartyNumber	9728324124
lastRedirectDn	22285
origCause_Value	16
dest_CauseValue	0
lastRedirectRedirectReason	0
lastRedirectRedirectOnBehalfOf	0
origTerminationOnBehalfOf	
destTerminationOnBehalfOf	
joinOnBehalfOf	0
duration	80

- **Mobility HandIn Example** - A dual-mode phone with the Enterprise number of 22285 and the cell number of 9728324124 establishes to the cell phone 9728324214. They talk for 39 seconds; then, the dual-mode phone gets carried into the Enterprise network, and the call gets switched from the cell network to the Enterprise network. The parties continue to talk for another 15 seconds.

Field Names	Call to cell #9728324214 CDR	HandIn Call to the Enterprise CDR
globalCallID_callId	864	864
origLegCallIdentifier	22481083	22481083
destLegCallIdentifier	22481085	22481087
callingPartyNumber	22202	22202
originalCalledPartyNumber	919728324124	22285
finalCalledPartyNumber	919728324124	22285
lastRedirectDn	919728324124	22285
origCause_Value	393216	0
dest_CauseValue	393216	16
lastRedirectRedirectReason	0	303
lastRedirectRedirectOnBehalfOf	0	24
origTerminationOnBehalfOf	24	24
destTerminationOnBehalfOf	24	12
joinOnBehalfOf	0	24
duration	39	15

- **Mobility HandOut Example** - A dual-mode phone has the Enterprise number of 22285 and the cell number of 9728324124. The handout number (H-number) specifies 555123. A call goes to the Enterprise number 22285. They talk for 21 seconds; then, the dual-mode phone gets carried out of the Enterprise network and into the cell network. The call gets switched from the Enterprise network to the cell network (9728324124). The parties continue to talk for another 39 seconds.

Field Names	Enterprise Call to 22285 CDR	Server Call from cell phone to H-Number CDR	Handout Call CDR
globalCallID_callId	964	965	964
origLegCallIdentifier	22481083	22481095	22481093
destLegCallIdentifier	22481094	22481096	22481095
callingPartyNumber	22202	9728324124	22202
originalCalledPartyNumber	22285	555123	9728324124
finalCalledPartyNumber	22285	555123	9728324124
lastRedirectDn	22285	555123	9728324124
origCause_Value	393216	393216	0
dest_CauseValue	393216	393216	16
lastRedirectRedirectReason	0	0	319
lastRedirectRedirectOnBehalfOf	0	0	24
origTerminationOnBehalfOf	24	24	24

destTerminationOnBehalfOf	24	24	12
joinOnBehalfOf	0	0	24
duration	21	0	39

- **Mobility Call Pickup Example** - A dual-mode phone with the Enterprise number of 22285 and the cell number of 9728324124, establishes a call to the Enterprise number 22285. They talk for 40 seconds; then, call pickup gets invoked. The call gets switched from the Enterprise phone to the cell phone. The parties continue to talk for another 111 seconds.

Field Names	Enterprise Call to 22285 CDR	Server Call to Cell Phone CDR	Final Handout Call CDR
globalCallID_callId	555	566	964
origLegCallIdentifier	22481111	22481222	22481111
destLegCallIdentifier	22481112	22481223	22481222
callingPartyNumber	22202		22202
originalCalledPartyNumber	22285	9728324124	9728324124
finalCalledPartyNumber	22285	9728324124	9728324124
lastRedirectDn	22285	9728324124	9728324124
origCause_Value	393216	0	0
dest_CauseValue	393216	0	16
lastRedirectRedirectReason	0	0	335
lastRedirectRedirectOnBehalfOf	0	0	24
origTerminationOnBehalfOf	24	24	24
destTerminationOnBehalfOf	24	24	12
joinOnBehalfOf	0	0	24
duration	40	0	111

- **Mobility IVR Example** - A call comes into the Cisco Unified Communications Manager with string (DID#RemoteDest#TargetNum#). The call gets redirected to the TargetNum. 9728131234 calls into an IVR, and data gets collected. The target destination specifies 812345 and the call gets redirected to 812345. The call remains connected for 60 seconds.

Field Names	Redirected Call CDR
globalCallID_callId	12345
origLegCallIdentifier	16677100
destLegCallIdentifier	16677102
callingPartyNumber	9728131234
originalCalledPartyNumber	8005559876
finalCalledPartyNumber	812345
lastRedirectDn	8005559876

origCause_Value	0
dest_CauseValue	16
lastRedirectRedirectReason	399
lastRedirectRedirectOnBehalfOf	24
origTerminationOnBehalfOf	0
destTerminationOnBehalfOf	0
duration	60

Intercom Calls

The following two examples show CDRs for intercom.

Intercom Examples

- **Whisper Intercom Example** - Phone 20000 invokes the intercom. The configured intercom partition name specifies “Intercom.”

Field Names	Original Call CDR
globalCallID_callId	1111000
origLegCallIdentifier	21822467
destLegCallIdentifier	21822468
callingPartyNumber	20000
originalCalledPartyNumber	20001
finalCalledPartyNumber	20001
origCause_Value	16
dest_CauseValue	0
origMediaTransportAddress_IP	0
origMediaTransportAddress_Port	0
destMediaTransportAddress_IP	-47446006
destMediaTransportAddress_Port	28480
origCalledPartyNumberPartition	Intercom
callingPartyNumberPartition	Intercom
finalCalledPartyNumberPartition	Intercom
duration	5

- **Talk-Back Intercom Example** - Phone 20000 presses the intercom button. 20001 invokes Talk-Back and talks to 20000. The configured intercom partition name specifies “Intercom.”

Field Names	Original Call CDR
globalCallID_callId	1111000

origLegCallIdentifier	21822469
destLegCallIdentifier	21822470
callingPartyNumber	20000
originalCalledPartyNumber	20001
finalCalledPartyNumber	20001
origCause_Value	16
dest_CauseValue	0
origMediaTransportAddress_IP	-131332086
origMediaTransportAddress_Port	29458
destMediaTransportAddress_IP	-47446006
destMediaTransportAddress_Port	29164
origCalledPartyNumberPartition	Intercom
callingPartyNumberPartition	Intercom
finalCalledPartyNumberPartition	Intercom
duration	5

CDR Field Descriptions

Table 10-4 defines all fields in the current CDRs in the order in which they appear in the CDR.

Table 10-4 CDR Field Descriptions

Field Name	Range of Values	Description
cdrRecordType	0, 1, 2	Defines the type of record. The following valid values apply: • 0—Start call detail record (not used) • 1—End call detail record (CDR) • 2—CMR record Default - For CDRs, this field always remains 1.
globalCallID_callManagerId	Positive Integer	Designates a unique Cisco Unified Communications Manager identity. The Global Call ID comprises two fields: globalCallID_callId globalCallID_callManagerId All records that are associated with a standard call have the same Global Call ID in them. Default -Ensure this field always is populated.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
globalCallID_callId	Positive Integer	Designates unique call identity value that is assigned to each call. The system allocates this identifier independently on each call server. Values get chosen sequentially when a call begins. A value gets assigned for each call, successful or unsuccessful. When Cisco Unified Communications Manager restarts, this values resets to 1. The Global Call ID consists of two fields: globalCallID_callId globalCallID_callManagerId All records that are associated with a standard call have the same Global Call ID in them. Default - Ensure this field always is populated.
origLegCallIdentifier	Positive Integer	Identifies the originating leg of a call. If the leg of a call persists across several sub-calls, and consequently several CDRs (as during a call transfer), this value remains constant. Default - Ensure this field always is populated.
dateTimeOrigination	Integer	Identifies the date and time when the user goes off hook or the date and time that the H.323 Setup message is received for an incoming call. The time gets stored as UTC. Default - Ensure this field always is populated.
origNodeId	Positive Integer	Identifies the server where the originator of the call is registered at the time that the call is made. Default - Ensure this field always is populated.
origSpan	0, Positive Integer	For calls that originate at a gateway, this field indicates the B-channel number of the T1, PRI, or BRI trunk where the call originates, or a zero value for FXS or FXO trunks. For H.323 gateways, the span number is unknown, and this field contains the call leg ID of the originator. For calls that did not originate at a gateway, the value specifies zero. Default - Populated based on these rules.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
origIpAddr	Integer	Identifies the IP address of the device that originated the call signaling. For Cisco Unified IP Phones, this field specifies the address of the phone. For PSTN calls, this field specifies the address of the H.323 gateway. The “IP Addresses” section on page 10-8 describes the IP address format. Default - Populated based on these rules.
callingPartyNumber	Text String	Specifies numeric string of up to 25 characters. For calls that originate at a Cisco Unified IP Phone, this field shows the extension number of the line that is used. For incoming H.323 calls, this field specifies the value that is received in the Calling Party Number field in the Setup message. This field reflects any translations that are applied to the Calling Party Number before it arrives at the Cisco Unified Communications Manager (such as translations at the gateway). For server calls, where Cisco Unified Communications Manager originates a half call without a calling party, this field may remain empty. CallingPartyNumber could contain a SIP URI. Default - Populated based on these rules.
callingPartyUnicodeLoginUserID	Unicode – UTF_8	Calling party login user ID. The format of this field specifies UTF_8. Default - Empty string “ “. If the user ID does not exist, this field stays empty.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
origCause_location	0 to 15	For clearing causes that are received over ISDN signaling links, specifies the Location field that is indicated in the ISDN release message. The "Call Termination Cause Codes" section on page 10-109 lists the valid values per Q.850. For clearing causes that are created internally by the Cisco Unified Communications Manager, this value specifies zero. Default - 0.
origCause_value	0 to 129	For calls cleared by the originating party, this field reflects the reason for clearance. Cisco Unified Communications Manager currently uses the Q.850 codes and some Cisco Unified Communications Manager defined codes. The "Call Termination Cause Codes" section on page 10-109 lists these. Some nonstandard cause codes changed in this release. For calls that are cleared by the terminating party, this field specifies zero. In addition to the standard values that are described in Q.850, when a call is split by a feature (transfer/conference), the CDR terminates, and this field gets set to 393216. This represents a proprietary value for this field. Default - 0.
origPrecedenceLevel	0 to 4	For MLPP, each call leg includes a precedence level. This field represents the precedence level of the original leg. • Precedence 0 = FLASH OVERRIDE/ EXECUTIVE OVERRIDE • Precedence 1 = FLASH • Precedence 2 = IMMEDIATE • Precedence 3 = PRIORITY • Precedence 4 = ROUTINE Default - 4.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
origMediaTransportAddress_IP	0, Integer	Identifies the IP address of the device that originates the media for the call. For Cisco Unified IP Phones, this field specifies the address of the phone. For PSTN calls, this field specifies the address of the H.323 gateway. The "IP Addresses" section on page 10-8 describes the IP address format. Default - 0. If media is not established, this field stays 0.
origMediaTransportAddress_Port	0, Positive Integer	Identifies the IP port number that is associated with the OrigMediaTransportAddress_IP field. Default - 0. If media is not established, this field stays 0.
origMediaCap_payloadCapability	0, Positive Integer	Identifies the codec type that the originator uses to transmit media. Cisco Unified Communications Manager currently uses the following payload capability values (0, 1-16, 18-20, 25, 32, 33, 81-86). The "Codec Types" section on page 10-108 lists the valid values. Default - 0. If media is not established, this field stays 0.
origMediaCap_maxFramesPerPacket	0, Positive Integer	Identifies the number of milliseconds of data per packet that is sent by the originating party. This field is normally gets set to 10, 20, or 30 for G.729 or G.711 codecs, but the field can store any nonzero value. Default - 0. If media is not established, this field stays 0.
origMediaCap_g723BitRate	0	This field is not used in the current release of Cisco Unified Communications Manager. This field will remain 0.
origVideoCap_Codec	0, 100 = H.261, 101 = H.263, 102 = Vieo	Identifies the codec type that the originator uses to transmit video (H.261, H.263, or Vieo.) Default - 0. If media is not established, this field stays 0.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
origVideoCap_Bandwidth	0, Positive Integer	Identifies the bandwidth measured in units of kbps. Default - 0. If media is not established, this field stays 0.
origVideoCap_Resolution	0, 1 = SQCIF, 2 = QCIF, 3 = CIF, 4 = CIF4, 5 = CIF16	Identifies the video resolution. Default - 0. If media is not established, this field stays 0.
origVideoTransportAddress_IP	0, Integer	Identifies the IP address of the device that originates the call. Default - 0. If media is not established, this field stays 0.
origVideoTransportAddress_Port	0, Positive Integer	Identifies the video RTP port that is associated with the origVideoTransportAddress_IP field. Default - 0. If media is not established, this field stays 0.
origRSVPAudioStat	0 to 5	Status of RSVP audio reservation from originator to terminator. 0 – No reservation. 1 – RSVP Reservation Failure condition at call setup or feature invocation. 2 – RSVP Reservation Success condition at call setup or feature invocation. 3 – RSVP Reservation No Response (RSVP Agent) condition at call setup or feature invocation. 4 – RSVP Mid Call Failure Preempted condition (preempted after call setup). 5 – RSVP Mid Call Failure Lost Bandwidth condition (includes all mid-call failures except MLPP preemption). Default – “0”.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
origRSVPVideoStat	0 to 5	Status of RSVP video reservation from originator to terminator. 0 – No reservation. 1 – RSVP Reservation Failure condition at call setup or feature invocation. 2 – RSVP Reservation Success condition at call setup or feature invocation. 3 – RSVP Reservation No Response (RSVP Agent) condition at call setup or feature invocation. 4 – RSVP MID Call Failure Preempted condition (preempted after call setup). 5 – RSVP MID Call Failure Lost Bandwidth condition (includes all mid-call failures except MLPP preemption). Default – “0”.
destLegCallIdentifier	0, Positive Integer	Identifies the terminating leg of a call. If the leg of a call persists across several sub-calls and, consequently, several CDRs (as during a call transfer), this value remains constant. Default - 0. If the destination cannot be reached, this field stays 0.
destNodeId	0, Positive Integer	Identifies the location where the terminating party of the call is registered at the time that the call is made. Default - 0. If the destination cannot be reached, this field stays 0.
destSpan	0, Positive integer	For calls that are received at a gateway, this field indicates the B channel number of the T1, PRI, or BRI trunk where the call is received, or a zero value for FXS or FXO trunks. For H.323 gateways, the span number remains unknown, and this field contains the call leg ID of the destination. For calls not terminating at a gateway, the value specifies zero. Default - 0. If the destination cannot be reached, this field stays 0.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
destIpAddr	0, Integer	Identifies the IP address of the device that terminates the call signaling. For Cisco Unified IP Phones, this field specifies the address of the phone. For PSTN calls, this field specifies the address of the H.323 gateway. The "IP Addresses" section on page 10-8 describes the IP address format. Default - 0. If the destination cannot be reached, this field stays 0.
originalCalledPartyNumber	Text String	Specifies number to which the original call was presented, prior to any call forwarding. If translation rules are configured, this number reflects the called number after the translations have been applied. Numeric string which could be either digits or a SIP URL. Default - empty string "". If destination cannot be reached, this field stays empty.
finalCalledPartyNumber	Text String	Specifies number to which the call finally gets presented, until it is answered or rings out. If no forwarding occurred, this number shows same number as originalCalledPartyNumber. For calls to a conference bridge, this field contains the actual identifier of the conference bridge, which is an alphanumeric string (for example, "b0019901001"). Numeric string that could be either digits or a SIP URL. Default - empty string "". If destination cannot be reached, this field stays empty.
finalCalledPartyUnicodeLoginUserID	Unicode – UTF_8	Final called party specifies login user ID. The format of this field is UTF_8. Default - Empty string " ". If the user ID does not exist, this field stays empty.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
destCause_location	0 to 15	For clearing causes that are received over ISDN signaling links, the ISDN release message indicates this location field. The "Call Termination Cause Codes" section on page 10-109 lists the valid values per Q.850. Some of the nonstandard cause codes changed in this release. For clearing causes that Cisco Unified Communications Manager creates internally, this value equals zero. Default - 0. If the destination cannot be reached, this field stays 0.
destCause_value	0 to 129	For calls that the destination party cleared, this field reflects the reason for the clearance. The "Call Termination Cause Codes" section on page 10-109 lists the valid values per Q.850. Some of the nonstandard cause codes changed in this release. For calls that the originating party clears, this field stays zero. In addition to the standard values that are described in Q.850, when a call gets split by a feature (transfer/conference), the CDR terminates, and this field gets set to 393216. This represents a proprietary value for this field. Default - 0. If the destination cannot be reached, this field stays 0.
destPrecedenceLevel	0 to 4	For MLPP, each call leg has a precedence level. This field represents the destination legs precedence level. • Precedence 0 = FLASH OVERRIDE • Precedence 1 = FLASH • Precedence 2 = IMMEDIATE • Precedence 3 = PRIORITY • Precedence 4 = ROUTINE Default - 4

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
destMediaTransportAddress_IP	0, Integer	Identifies the IP address of the device that terminates the media for the call. For Cisco Unified IP Phones, this field designates the address of the phone. For PSTN calls, this field designates the address of the H.323 gateway. The "IP Addresses" section on page 10-8 describes the IP address format. Default - 0. If the destination cannot be reached, this field stays 0.
destMediaTransportAddress_Port	0, Positive Integer	Identifies the IP port number that is associated with the DestMediaTransportAddress_IP field. Default - 0. If the destination cannot be reached, this field stays 0.
destMediaCap_payloadCapability	0, Positive Integer	Identifies the codec type that the terminating party used to transmit media. Cisco Unified Communications Manager currently uses the following payload capability values: 0, 1-16, 18-20, 25, 32, 33, 81-86. The "Codec Types" section on page 10-108 lists the valid values. Default - 0. If the destination cannot be reached, this field stays 0.
destMediaCap_maxFramesPerPacket	0, Positive Integer	Identifies the number of milliseconds of data per packet that the terminating party of the call sent. This field normally gets set to 10, 20, or 30 for G.729 or G.711 codecs but can store any nonzero value. This field can specify zero if the media is never established. Default - 0. If the destination cannot be reached, this field stays 0.
destMediaCap_g723BitRate	0	This field is not used in the current release of Cisco Unified Communications Manager. Default - This field stays 0.
destVideoCap_Codec	0, 100 = H.261, 101 = H.263, 102 = Vieo	Identifies the codec type that the terminating party used to transmit video (H.261, H.263, or Vieo). Default - 0. If the destination cannot be reached, this field stays 0.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
destVideoCap_Bandwidth	0, Positive Integer	Identifies the bandwidth measured in units of kbps. Default - 0. If the destination cannot be reached, this field stays 0.
destVideoCap_Resolution	0, 1 = SQCIF, 2 = QCIF, 3 = CIF, 4 = CIF4, 5 = CIF16	Identifies the video resolution. Default - 0. If the destination cannot be reached, this field stays 0.
destVideoTransportAddress _IP	0, Integer	Identifies the IP address of the device that receives the call. Default - 0. If the destination cannot be reached, this field stays 0.
destVideoTransportAddress_Port	0, Positive Integer	Identifies the video RTP port that is associated with the destVideoTransportAddress_IP field. Default - 0. If the destination cannot be reached, this field stays 0.
destRSVPAudioStat	0 - 5	Status of RSVP audio reservation from terminator to originator. 0 – No reservation. 1 – RSVP Reservation Failure condition at call setup or feature invocation. 2 – RSVP Reservation Success condition at call setup or feature invocation. 3 – RSVP Reservation No Response (RSVP Agent) condition at call setup or feature invocation. 4 – RSVP Mid Call Failure Preempted condition (preempted after call setup). 5 – RSVP Mid Call Failure Lost Bandwidth condition (includes all mid call failures except MLPP preemption). Default – “0”

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
destRSVPVideoStat	0 - 5	Status of RSVP video reservation from terminator to originator. 0 – No reservation. 1 – RSVP Reservation Failure condition at call setup or feature invocation. 2 – RSVP Reservation Success condition at call setup or feature invocation. 3 – RSVP Reservation No Response (RSVP Agent) condition at call setup or feature invocation. 4 – RSVP Mid Call Failure Preempted condition (preempted after call setup). 5 – RSVP Mid Call Failure Lost Bandwidth condition (includes all mid call failures except MLPP preemption). Default – “0”
dateTimeConnect	0, Integer	Identifies the date and time that the call connects. The time gets stored as UTC. If the call is never answered, this value shows zero. Default - 0. If the call is never connected, this field stays 0.
dateTimeDisconnect	0, Integer	Identifies the date and time when the call is cleared. This field gets set even if the call never connects. The time gets stored as UTC. Default - 0. If the call is never connected, this field stays 0.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
lastRedirectDn	Text String	Specifies a numeric string of up to 25 characters. Numeric string can hold the digits or a SIP URL. For forwarded calls, this field specifies the phone number of the next to last hop before the call reaches its final destination. If only one hop occurs, this number matches the OriginalCalledPartyNumber. For calls that are not forwarded, this field matches the OriginalCalledPartyNumber and the FinalCalledPartyNumber. For calls to a conference bridge, this field contains the actual identifier of the conference bridge, which is an alphanumeric string (for example, “b0019901001”). Default - empty string “”. If call was never redirected, this field remains empty.
pkid	Text String	Identifies a text string that the database uses internally to uniquely identify each row. This text string provides no meaning to the call itself. Default - A unique ID should always populate this field.
originalCalledPartyNumberPartition	Text String	Uniquely identifies the partition name that is associated with the OriginalCalledPartyNumber field because Cisco Unified Communications Manager supports multiple Cisco Unified IP Phones with the same extension number in different partitions. For calls that egress through an H.323 gateway, this field uniquely specifies the partition name that is associated with the route pattern that points to the gateway. Default - empty string “”. If the original called party does not have a partition, this field remains empty.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
callingPartyNumberPartition	Text String	Uniquely identifies the partition name that is associated with the CallingPartyNumber field because Cisco Unified Communications Manager supports multiple Cisco Unified IP Phones with the same extension number in different partitions. For calls that ingress through an H.323 gateway, this field remains blank. Default - empty string “”. If the original called party does not have a partition, this field remains empty.
finalCalledPartyNumberPartition	Text String	Uniquely identifies the partition name that is associated with the FinalCalledPartyNumber field because Cisco Unified Communications Manager supports multiple Cisco Unified IP Phones with the same extension number in different partitions. For calls that egress through an H.323 gateway, this field uniquely specifies the partition name that is associated with the route pattern that points to the gateway. Default - empty string “”. If the final called party does not have a partition, this field remains empty.
lastRedirectDnPartition	Text String	Uniquely identifies the partition name that is associated with the LastRedirectDn field because Cisco Unified Communications Manager supports multiple Cisco Unified IP Phones with the same extension number in different partitions. For calls that egress through an H.323 gateway, this field specifies the partition name that is associated with the route pattern that points to the gateway. Default - empty string “”. If the last redirecting Party does not have a partition or the call was never redirected, this field stays empty.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
duration	0, Positive integer	Identifies the difference between the Connect Time and Disconnect Time. This field specifies the time that the call remains connected, in seconds. This field remains zero if the call never connects or if it connects for less than 1 second. Default - 0.
origDeviceName	Text String	Specifies text string that identifies the name of the originating device. Default - Ensure this field always is populated.
destDeviceName	Text String	Specifies text string that identifies the name of the destination device. Default - empty string ““. If the original device does not have a name, this field stays empty.
origCallTerminationOnBehalfOf	0, Positive Integer	Specifies code that identifies why the originator was terminated. For example, if the originator of the call hangs up the phone, the OnBehalfOf code shows “12” for Device. If the call terminates because of a transfer, the OnBehalfOf code shows “10” for Transfer. See the “OnBehalfof Codes” section on page 10-114 for a list of the codes. This release added new OnBehalfOf codes. Default - 0.
destCallTerminationOnBehalfOf	0, Positive Integer	Specifies code that identifies why the destination was terminated. For example, if the originator of the call hangs up the phone, the OnBehalfOf code shows “12” for Device. If the call terminates because of a transfer, the OnBehalfOf code shows “10” for Transfer. See the “OnBehalfof Codes” section on page 10-114 for a list of the codes. This release added new OnBehalfOf codes. Default - 0.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
origCalledPartyRedirectOnBehalfOf	0, Positive Integer	Specifies code that identifies the reason for redirection of the original called party. For example, if the original called party was redirected because of a conference, the OnBehalfOf code specifies "4." See the "OnBehalfof Codes" section on page 10-114 for a list of the codes. This release added new OnBehalfOf codes. Default - 0.
lastRedirectRedirectOnBehalfOf	0, Integer	Specifies code that identifies the reason for redirection of the last redirected party. For example, if the last redirected party was redirected on behalf of a conference, the OnBehalfOf code specifies "4." See the "OnBehalfof Codes" section on page 10-114 for a list of the codes. This release added new OnBehalfOf codes. Default - 0.
origCalledPartyRedirectReason	0, Integer	Identifies the reason for a redirect of the original called party. See the "Redirect Reason Codes" section on page 10-113 for a complete list of the codes. This release added new redirect reason values. Default - 0.
lastRedirectRedirectReason	0, Integer	Identifies the last redirect reason for redirection. See the "Redirect Reason Codes" section on page 10-113 for a complete list of the codes. This release added new redirect reason values. Default: 0.
destConversationID	0, Integer	Specifies unique identifier that is used to identify the parties of a conference call. For conference chaining scenarios, the origConversationID and destConversationID fields identify which conferences are chained together. Default: 0.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
globalCallId_ClusterId	Text String	Unique ID that identifies a cluster of Cisco Unified Communications Manager. The field is generated at installation and is not used by Cisco Unified Communications Manager. The fields globalCallId_ClusterId + globalCallId_CMId + globalCallId_CallId make up this unique key. Default: This field should always be populated.
joinOnBehalfOf	0, Integer	Specifies code that identifies the reason for a join. For example, if the join takes place on behalf of a transfer, the OnBehalfOf code specifies "10." See the "OnBehalfof Codes" section on page 10-114 for a list of the codes. This release added new OnBehalfOf codes. Default: 0
Comment	Text String	Allows features to add text to the CDRs. This text can describe details about the call. For example, the following field flags malicious calls: Tag—CallFlag Value—MALICIOUS Default: Empty string "".
authCodeDescription	Text String	Description of the FAC. Default: Empty string "" or null.
authorizationLevel	0, Integer	Level of the FAC. Default: 0
clientMatterCode	Text String	Before the system extends a call, the user enters a client matter code that can be used for assigning account or billing codes to calls. Default: Empty string "" or null.

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
origDTMFMethod	0, Positive Integer	DTMF method that the originator uses. 0 - No DTMF - Use ANY matched DTMF. 1 - OOB - Use OOB if endpoints behind SIPTrunk support it. 2 - 2833 - Use RFC2833 if endpoints behind SIPTrunk support it. 3 - OOB and 2833 - Use both KPML and RFC2833 if endpoints behind SIPTrunk can support both. 4 - Unknown Default: 0 (No preference)
destDTMFMethod	0, Positive Integer	DTMF method that the destination uses. 0 - No DTMF - Use ANY matched DTMF. 1 - OOB - Use OOB if endpoints behind SIPTrunk support it. 2 - 2833 - Use RFC2833 if endpoints behind SIPTrunk support it. 3 - OOB and 2833 - Use both KPML and RFC2833 if endpoints behind SIPTrunk can support both. 4 - Unknown. Default: 0 (No preference)
callSecuredStatus	0, Positive Integer	The highest security status that is reached during a call. For example, if the call is originally unsecured, then later the call changes to secured, the CDR contains 1 for “Secured” even though different portions of the call had different status values. 0 - Non-secured 1 - Authenticated (not encrypted) 2 - Secured (encrypted) Default: 0 (Non-secured)
origConversationID	Integer	Identifies the conference ID that is associated with the originating leg of the call. In most cases, this field equals 0. For conference chaining scenarios, the origConversationID and destConversationID fields identify which conferences are chained together. Default: 0

Table 10-4 CDR Field Descriptions (continued)

Field Name	Range of Values	Description
origMediaCap_Bandwidth	0, Positive Integer	The media bandwidth that is used at the origination of the call. Default: 0
destMediaCap_Bandwidth	0, Positive Integer	The media bandwidth used at the destination of the call. Default: 0
authorizationCodeValue	Text String	Forced Authorization Code (FAC) that is associated with the call. Default: Empty string "" or null.

CMR Field Descriptions (Diagnostic)

Table 10-5 contains the fields, range of values, and field descriptions of the CMRs in the order in which they appear in the CMR.

Table 10-5 CMR Field Descriptions

Field Name	Range of Values	Description
cdrRecordType	0, 1, or 2	Specifies the type of this specific record. The following valid values apply: • 0—Start call detail record (not used) • 1—End call detail record • 2—CMR record Default - For CMRs, this field always specifies 2.
globalCallID_callManagerId	Positive Integer	Specifies a unique Cisco Unified Communications Manager identity. This field makes up half of the Global Call ID. The Global Call ID comprises the following fields: • globalCallId_callId • globalCallID_callManagerID All records that are associated with a standard call have the same Global Call ID in them. Default - Ensure this field always is populated.

Table 10-5 CMR Field Descriptions (continued)

Field Name	Range of Values	Description
globalCallId_callId	Positive Integer	Specifies a unique call identity value that gets assigned to each call. The system allocates this identifier independently on each call server. Values get chosen sequentially when a call begins. Each call, successful or unsuccessful, receives value assignment. This field makes up half the Global Call ID. The Global Call ID comprises the following two fields: • globalCallId_callId • globalCallID_callManagerID All records that are associated with a standard call have the same Global Call ID in them. Default - Ensure this field always is populated.
nodeId	Positive Integer	Specifies the server where this record was generated. Default - Ensure this field always is populated.
callIdentifier	Positive Integer	Identifies the call leg to which this record pertains. Default - Ensure this field always is populated.
directoryNumber	Integer	Specifies the directory number of the device from which these diagnostics were collected. Default - Ensure this field always is populated.
dateTimeStamp	Integer	Represents the approximate time that the device went on hook. Cisco Unified Communications Manager records the time when the phone responds to a request for diagnostic information. Default - Ensure this field always is populated.

Table 10-5 CMR Field Descriptions (continued)

Field Name	Range of Values	Description
numberPacketsSent	Integer	Designates the total number of Routing Table Protocol (RTP) data packets that the device transmitted since starting transmission on this connection. The value remains zero if the connection was set in “receive only” mode. Default - 0.
numberOctetsSent	Integer	Specifies the total number of payload octets (that is, not including header or padding) that the device transmitted in RTP data packets since starting transmission on this connection. The value remains zero if the connection was set in “receive only” mode. Default - 0.
numberPacketsReceived	Integer	Specifies the total number of RTP data packets that the device received since starting reception on this connection. The count includes packets that are received from different sources if this is a multicast call. The value remains zero if the connection was set in “send only” mode. Default - 0.
numberOctetsReceived	Integer	Specifies the total number of payload octets (that is, not including header or padding) that the device received in RTP data packets since starting reception on this connection. The count includes packets that are received from different sources if this is a multicast call. The value remains zero if the connection was set in “send only” mode. Default - 0.

Table 10-5 CMR Field Descriptions (continued)

Field Name	Range of Values	Description
numberPacketsLost	Integer	Designates the total number of RTP data packets that have been lost since the beginning of reception. This number designates the number of packets that were expected, less the number of packets that were actually received, where the number of packets that were received includes any that are late or duplicates. Thus, packets that arrive late do not get counted as lost, and the loss may be negative if duplicates exist. The number of packets that are expected designates the extended last sequence number that was received, as defined next, less the initial sequence number that was received. The value remains zero if the connection was set in "send only" mode. For detailed information, see RFC 1889. Default - 0.
jitter	Integer	Provides an estimate of the statistical variance of the RTP data packet interarrival time, measured in milliseconds and expressed as an unsigned integer. The interarrival jitter J specifies the mean deviation (smoothed absolute value) of the difference D in packet spacing at the receiver, compared to the sender for a pair of packets. RFC 1889 contains detailed computation algorithms. The value remains zero if the connection was set in "send only" mode. Default - 0.
latency	Integer	Designates value that is an estimate of the network latency, expressed in milliseconds. This value represents the average value of the difference between the NTP timestamp that the RTP Control Protocol (RTCP) messages indicates and the NTP timestamp of the receivers, measured when these messages are received. Cisco Unified Communications Manager obtains the average by summing all estimates then dividing by the number of RTCP messages that have been received. For detailed information, see RFC 1889. Default - 0.

Table 10-5 CMR Field Descriptions (continued)

Field Name	Range of Values	Description
pkid	Text String	Identifies a text string that the database uses internally to uniquely identify each row. This text string provides no meaning to the call itself. Default - The system always populates this field with a unique ID.
directoryNumberPartition	Text String	Identifies the partition of the directory number. Default - Empty string, “”. This field may remain empty if no partition exists.
deviceName	Text String	Identifies the name of the device. Default - Empty string “”. This field may remain empty if there is no device name.
globalCallId_ClusterId	Text String	Designates unique ID that identifies a single Cisco Unified Communications Manager. The system generates this field during installation, but Cisco Unified Communications Manager does not use it: globalCallId_ClusterId + globalCallId_callId. Default - Ensure this field always is populated.
varVQMetrics	Text String	This field contains a variable number of voice quality metrics. This field comprises a string of voice quality metrics that are separated by a semicolon. The format of the string follows: fieldName=value;fieldName=value.precision This example shows voice quality data, but the names may differ. "MLQK=4.5000;MLQKav=4.5000;MLQKmn=4.5000;MLQKmx=4.5000;MLQKvr=0.95;CCR=0.0000;ICR=0.0000;ICRmx=0.0000;CS=0;SCS=0” **Note** See Table 10-6 “K-Factor Data Stored in Cisco Unified Communications Manager CMRs” for a complete list of K-Factor data.

K-Factor Data in CMRs

K-factor, an endpoint mean opinion score (MOS) estimation algorithm that is defined in ITU standard P.VTQ, serves as a general estimator and is used to estimate the mean value of a perceptual evaluation of speech quality (PESQ) population for a specific impairment pattern.

MOS relates to the output of a well designed listening experiment. All MOS experiments use a five point PESQ scale as defined in ITU standard P.862.1, which describes the PESQ as an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs.

Be aware that the MOS estimate is a number that is inversely proportional to frame loss density. Clarity decreases as more frames are lost or discarded at the receiving end. The loss or discarding of these frames represents "concealment." Concealment statistics measure packet (frame) loss and its effect on voice quality in an impaired network.

K-factor represents a weighted estimate of average user annoyance due to distortions that are caused by effective packet loss such as dropouts and warbles. It does not estimate the impact of delay-related impairments such as echo. It provides an estimate of listening quality (MOS-LQO) rather than conversational quality (MOS-CQO), and measurements of average user annoyance range from 1 (poor voice quality) to 5 (very good voice quality).

Because K-factor is trained or conditioned by speech samples from numerous speech databases, where each training sentence or network condition associated with a P.862.1 value has a duration of 8 seconds, for more accurate scores, the system generates k-factor estimates for every 8 seconds of active speech.

Consider K-factor and other MOS estimators to be secondary or derived statistics because they warn a network operator of frame loss only after the problem becomes significant. Packet counts, concealment ratios, and concealment second counters represent primary statistics because they alert the network operator before network impairment has an audible impact or is visible through MOS.

Table 10-6 displays the K-factor date that is stored in the Cisco Unified Communications Manager CMRs.

Table 10-6 K-Factor Data Stored in Cisco Unified Communications Manager CMRs

Field Name	Phone Display Name	D&I User Interface Text and Description
CCR	Cum Conceal Ratio	Cumulative Conceal Ratio specifies the cumulative ratio of concealment time over speech time that is observed after starting a call.
ICR	Interval Conceal Ratio	Interval Conceal Ratio specifies an interval-based average concealment rate that is the ratio of concealment time over speech time for the last 3 seconds of active speech.
ICRmx	Max Conceal Ratio	Interval Conceal Ratio Max specifies the maximum concealment ratio that is observed during the call.
CS	Conceal Secs	Conceal Secs.specifies the duration of time during which some concealment is observed during a call.

Table 10-6 K-Factor Data Stored in Cisco Unified Communications Manager CMRs (continued)

Field Name	Phone Display Name	D&I User Interface Text and Description
SCS	Severely Conceal Secs	Severely Conceal Secs specifies the time during which a significant amount of concealment is observed. If the concealment that is observed is usually greater than 50 milliseconds or approximately 5 percent, the speech probably represents not very audible speech.
MLQK	MOS LQK	MOS Listening Quality K-factor provides an estimate of the MOS score of the last 9 seconds of speech on the reception signal path.
MLQKmn	Min MOS LQK	MOS Listening Quality K-factor Min specifies the minimum score that was observed since the beginning of a call and represents the worst sounding 8 second interval.
MLQKmx	Max MOS LQK	MOS Listening Quality K-factor Max specifies the maximum score that was observed since the beginning of a call and represents the best sounding 8 second interval.
MLQKav	Avg MOS LQK	MOS Listening Quality K-factor Avg8 specifies the running average of scores that were observed since the beginning of a call.

Codec Types

Table 10-7 contains the compression and payload types that may appear in the codec fields.

Table 10-7 Codec Types

Value	Description
1	NonStandard
2	G711Alaw 64k
3	G711Alaw 56k
4	G711mu-law 64k
5	G711mu-law 56k
6	G722 64k
7	G722 56k
8	G722 48k
9	G7231
10	G728
11	G729
12	G729AnnexA
13	Is11172AudioCap

Table 10-7 Codec Types (continued)

Value	Description
14	Is13818AudioCap
15	G.729AnnexB
16	G.729 Annex AwAnnexB
18	GSM Full Rate
19	GSM Half Rate
20	GSM Enhanced Full Rate
25	Wideband 256K
32	Data 64k
33	Data 56k
40	G7221 32K
41	G7221 24K
42	AAC
80	GSM
81	ActiveVoice
82	G726_32K
83	G726_24K
84	G726_16K
86	iLBC
100	H261
101	H263
102	Vieo
103	H264
106	H224

Call Termination Cause Codes

The following tables contain call termination cause codes that may appear in the Cause fields in CDRs.

- "Call Termination Cause Codes"
- "Cisco-Specific Call Termination Cause Codes"

Table 10-8 Call Termination Cause Codes

Code	Description
0	No error
1	Unallocated (unassigned) number
2	No route to specified transit network (national use)
3	No route to destination

Table 10-8 Call Termination Cause Codes (continued)

Code	Description
4	Send special information tone
5	Misdialed trunk prefix (national use)
6	Channel unacceptable
7	Call awarded and being delivered in an established channel
8	Preemption
9	Preemption—circuit reserved for reuse
16	Normal call clearing
17	User busy
18	No user responding
19	No answer from user (user alerted)
20	Subscriber absent
21	Call rejected
22	Number changed
26	Non-selected user clearing
27	Destination out of order
28	Invalid number format (address incomplete)
29	Facility rejected
30	Response to STATUS ENQUIRY
31	Normal, unspecified
34	No circuit/channel available
38	Network out of order
39	Permanent frame mode connection out of service
40	Permanent frame mode connection operational
41	Temporary failure
42	Switching equipment congestion
43	Access information discarded
44	Requested circuit/channel not available
46	Precedence call blocked
47	Resource unavailable, unspecified
49	Quality of Service not available
50	Requested facility not subscribed
53	Service operation violated
54	Incoming calls barred
55	Incoming calls barred within Closed User Group (CUG)
57	Bearer capability not authorized
58	Meet-Me secure conference minimum security level not met

Table 10-8 Call Termination Cause Codes (continued)

Code	Description
62	Inconsistency in designated outgoing access information and subscriber class
63	Service or option not available, unspecified
65	Bearer capability not implemented
66	Channel type not implemented
69	Requested facility not implemented
70	Only restricted digital information bearer capability is available (national use).
79	Service or option not implemented, unspecified
81	Invalid call reference value
82	Identified channel does not exist.
83	A suspended call exists, but this call identity does not.
84	Call identity in use
85	No call suspended
86	Call having the requested call identity has been cleared.
87	User not member of CUG (Closed User Group)
88	Incompatible destination
90	Destination number missing and DC not subscribed
91	Invalid transit network selection (national use)
95	Invalid message, unspecified
96	Mandatory information element is missing.
97	Message type nonexistent or not implemented
98	Message not compatible with the call state, or the message type nonexistent or not implemented
99	An information element or parameter non-existent or not implemented
100	Invalid information element contents
101	Message not compatible with the call state
102	Call terminated when timer expired; a recovery routine executed to recover from the error.
103	Parameter nonexistent or not implemented - passed on (national use)
110	Message with unrecognized parameter discarded
111	Protocol error, unspecified
122	Precedence Level Exceeded
123	Device not Preemptable
125	Out of bandwidth (Cisco specific)
126	Call split (Cisco specific)
127	Interworking, unspecified
129	Precedence out of bandwidth

Table 10-9 Cisco-Specific Call Termination Cause Codes

Code	Description
262144	Conference Full (was 124)
393216	Call split (was 126) This code applies when a call terminates during a transfer operation because it was split off and terminated (was not part of the final transferred call). This can help determine which calls terminated as part of a feature operation.
458752	Drop any party/drop last party (was 128)
16777257	CCM_SIP_400_BAD_REQUEST
33554453	CCM_SIP_401_UNAUTHORIZED
50331669	CCM_SIP_402_PAYMENT_REQUIRED
67108885	CCM_SIP_403_FORBIDDEN
83886081	CCM_SIP_404_NOT_FOUND
100663359	CCM_SIP_405_METHOD_NOT_ALLOWED
117440591	CCM_SIP_406_NOT_ACCEPTABLE
134217749	CCM_SIP_407_PROXY_AUTHENTICATION_REQUIRED
150995046	CCM_SIP_408_REQUEST_TIMEOUT
184549398	CCM_SIP__410_GONE
201326719	CCM_SIP_411_LENGTH_REQUIRED
234881151	CCM_SIP_413_REQUEST_ENTITY_TOO_LONG
251658367	CCM_SIP_414_REQUEST_URI_TOO_LONG
268435535	CCM_SIP_415_UNSUPPORTED_MEDIA_TYPE
285212799	CCM_SIP_416_UNSUPPORTED_URI_SCHEME
83886207	CCM_SIP_420_BAD_EXTENSION
369098879	CCM_SIP_421_EXTENSION_REQUIRED
402653311	CCM_SIP_423_INTERVAL_TOO_BRIEF
1073741842	CCM_SIP_480_TEMPORARILY_UNAVAILABLE
1090519081	CCM_SIP_481_CALL_LEG_DOES_NOT_EXIST
1107296281	CCM_SIP_482_LOOP_DETECTED = 0x42000000 + EXCHANGE_ROUTING_ERROR
1124073497	CCM_SIP_483_TOO_MANY_HOOPS
1140850716	CCM_SIP_484_ADDRESS_INCOMPLETE
1157627905	CCM_SIP_485_AMBIGUOUS
1174405137	CCM_SIP_486_BUSY_HERE
1191182367	CCM_SIP_487_REQUEST_TERMINATED
1207959583	CCM_SIP_488_NOT_ACCEPTABLE_HERE
1258291217	CCM_SIP_491_REQUEST_PENDING
1291845649	CCM_SIP_493_UNDECIPHERABLE
1409286185	CCM_SIP_500_SERVER_INTERNAL_ERROR

Table 10-9 Cisco-Specific Call Termination Cause Codes (continued)

Code	Description
1442840614	CCM_SIP_502_BAD_GATEWAY
1459617833	CCM_SIP_503_SERVICE_UNAVAILABLE
1476395110	CCM_SIP__504_SERVER_TIME_OUT
1493172351	CCM_SIP_505_SIP_VERSION_NOT_SUPPORTED
1509949567	CCM_SIP_513_MESSAGE_TOO_LARGE
2701131793	CCM_SIP_600_BUSY_EVERYWHERE
2717909013	CCM_SIP_603_DECLINE
2734686209	CCM_SIP_604_DOES_NOT_EXIST_ANYWHERE
2751463455	CCM_SIP_606_NOT_ACCEPTABLE

Redirect Reason Codes

Table 10-10 contains the available Redirect Reason Codes that may appear in a record.

Table 10-10 Redirect Reason Codes

Q.931 Standard Redirect Reason Codes	
Value	**Description**
0	Unknown
1	Call Forward Busy
2	Call Forward No Answer
4	Call Transfer
5	Call Pickup
7	Call Park
8	Call Park Pickup
9	CPE Out of Order
10	Call Forward
11	Call Park Reversion
15	Call Forward All

Nonstandard Redirect Reason Codes	
18	Call Deflection
34	Blind Transfer
50	Call Immediate Divert
66	Call Forward Alternate Party
82	Call Forward On Failure
98	Conference

Table 10-10 Redirect Reason Codes (continued)

114	Barge
129	Aar
130	Refer
146	Replaces
162	Redirection (3xx)
177	SIP-forward busy greeting
207	Follow Me (SIP-forward all greeting)
209	Out of Service (SIP-forward busy greeting)
239	Time Of Day (SIP-forward all greeting)
242	Do Not Disturb (SIP-forward no answer greeting)
257	Unavailable (SIP-forward busy greeting)
274	Away (SIP-forward no answer greeting)
303	Mobility HandIn
319	Mobility HandOut
335	Mobility Cell Pickup
354	Recording
370	Monitoring
399	Mobility IVR

OnBehalfof Codes

Table 10-11 contains the available OnBehalfof Codes that may appear in a CDR record.

Table 10-11 OnBehalfof Codes

Value	Description
0	Unknown
1	CctiLine
2	Unicast Shared Resource Provider
3	Call Park
4	Conference
5	Call Forward
6	Meet-Me Conference
7	Meet-Me Conference Intercepts
8	Message Waiting
9	Multicast Shared Resource Provider
10	Transfer
11	SSAPI Manager

Table 10-11 OnBehalfof Codes (continued)

Value	Description
12	Device
13	Call Control
14	Immediate Divert
15	Barge
16	Pickup
17	Refer
18	Replaces
19	Redirection
20	Callback
21	Path Replacement
22	FacCmc Manager
23	Malicious Call
24	Mobility
25	Aar
26	Directed Call Park
27	Recording
28	Monitoring

Related Topics

- CDR Analysis and Reporting Overview, page 1-1
- Getting Started with CDR Analysis and Reporting, page 2-1
- CAR System Configuration, page 3-1
- CAR Report Configuration, page 4-1
- CAR User Reports Configuration, page 5-1
- CAR System Reports Configuration, page 6-1
- CAR Device Reports Configuration, page 7-1
- CDR Search Configuration, page 8-1
- Export CDR/CMR Records Configuration, page 9-1
- CAR Report Results, page 11-1

Related Documentation

The following documents contain additional information related to CDRs:

- *Cisco Unified Serviceability Administration Guide*
- *Cisco Communications Manager System Guide*