Cisco CallManager 4.2(1) Call Detail Record Definition

This document describes the format and logic of the call detail records (CDRs) that the Cisco CallManager Release 4.2(1) system generates. An integration partner can use this information for post-processing activities such as generating billing records and network analysis. This document describes how to access the database, how to interpret fields in the database schema, and some of the known issues.

When you install your system, the system specifies that call detail records (CDRs) are disabled by default. You can enable and disable CDR records at any time while the system is in operation. You do not need to restart the Cisco CallManager for changes to take effect. The system responds to all changes within a few seconds.

Contents

This document covers the following topics:

Corporate Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

New and Changed Information

This section describes any new features and or changes for CDRs that are pertinent to the specified release of Cisco CallManager.

Cisco CallManager Release 4.2(1)

For this release, you can configure Cisco CallManager to report the directory number (DN) of the hunt group member who answered a direct call to the hunt pilot number in the final called party number field in the CDR. Previously, Cisco CallManger reported the hunt pilot DN in the final called party number field in the CDR.

When you enable the service parameter and set Show Line Group Member DN in the finalCalledPartyNumber CDR field parameter to True, the finalCalledPartyNumber field in the CDR will record the DN of the member who answered the call. When you disable the service parameter and set Show Line Group Member DN in finalCalledPartyNumber CDR field parameter to False, the finalCalledPartyNumber field in the CDR will record the hunt pilot DN. By default, the system sets the Show Line Group Member DN in finalCalledPartyNumber CDR field parameter to False. For example, when a direct call to hunt pilot number of 8600 is routed and picked up by hunt group member with extension 2037, the CDR displays the final called party number (finalCalledPartyNumber) as 2037 when Show Line Group Member DN in finalCalledPartyNumber CDR field parameter is set to True. Whereas the CDR displays the final called party number (finalCalledPartyNUmber) as 8600 when Show Line Group Member DN in finalCalledPartyNumber CDR field parameter is set to False.

In addition to the existing voice quality data supported by the SCCP phone, new measurements are added in the new varVQMetrics field. This string field contains voice quality metrics separated by semicolons.

The format of the string is either fieldName=value or fieldName=value/precision.

Cisco CallManager Release 4.1(3)

For this release, the content of the CDR records changed for the new Auto Pickup feature, but no new CDR fields or changed CDR fields were added. Enhancements to the existing Call Pickup and Group Call Pickup features provide the Auto Pickup feature.

You can enable and disable Auto Pickup by using the new service parameter Auto Pickup Enabled. By default, the system sets the Auto Pickup Enabled parameter to False. When the parameter is set to True, Auto Pickup applies to all types of Call Pickup.

Auto Pickup

The following list gives the three types of auto pickup:

- Auto Call
- Auto Group
- Auto Other

The new Auto Pickup feature generates only two CDR records: one CDR for the ringing call and another CDR for the final connected call that is picked up. Both CDRs have the same Call ID.

For the first CDR, the origTerminationOnBehalfOf and destTerminationOnBehalfOf fields get set to 16 (Pickup), which indicates that the call terminated on behalf of the Pickup feature.

For the second CDR, the lastRedirectOnBehalfOf and joinOnBehalfOf fields get set to 16 (Pickup), which indicates that the system joined the call on behalf of the Pickup feature. The lastRedirectDn will indicate from where the call was picked up, that is, lastRedirectDn will contain the party that was ringing when the call was picked up. The lastRedirectRedirectReason will contain the redirect reason 5 (Pickup).

Pickup

The existing pickup features generate only one CDR record. The origCalledPartyRedirectOnBehalfOf, lastRedirectRedirectOnBehalfOf, and joinOnBehalfOf fields get set to 5 (Call Froward), which indicates that the Call Forward feature redirected the call. The origCalledPartyRedirectReason and lastRedirectRedirectReason contain the redirect reason code of 5 (Pickup).

Cisco CallManager Release 4.1(2)

The following list provides the features or changes for CDRs in Cisco CallManager release 4.1(2):

- Forced Authorization Codes (FAC)—Forces the user to enter a valid authorization code prior to extending calls to classes of dialed numbers, such as external calls, toll calls, and international calls. Authorization information gets written to the Cisco CallManager database.
- Client Matter Codes (CMC)—Before extending a call, Allows the user to enter a “client matter” code that the customer can use for assigning account or billing codes to calls that are placed. Client Matter Code information gets written to the Cisco CallManager CDR database.

The 4.1(2) Cisco CallManager release provides three new CDR fields to support FAC and CMC:

- authCodeDescription
- authorizationLevel
- clientMatterCode

Cisco CallManager Release 4.0(1)

The following list provides the features or changes for CDRs in Cisco CallManager release 4.0(1):

- Identifies Multilevel Precedence and Preemption (MLPP)
  - Adds the new field `origPrecendenceLevel` for the precedence level of the originating leg of the call
  - Adds the new field `destPrecedenceLvel` for the precedence level of the destination leg of the call
  - MLPP utilizes existing cause codes 8, 9, 46, 50, and 128
- Identifies malicious calls by adding a new `Comment` field
- Drop any party feature utilizes existing cause fields: `origCause_value` and `destCause_value`
- The `OnBehalfOf` field contains a new code (value = 14) for the Immediate Divert feature and value = 15 for Barge.
- The following new fields support video in Cisco CallManager:
  - `origVideoCap_Codec`
  - `destVideoCap_Codec`
  - `origVideoCap_Bandwidth`

  - destVideoCap_Bandwidth
  - origVideoCap_Resolution
  - destVideoCap_Resolution
  - origVideoTransportAddress_IP
  - origVideoTransportAddress_Port
  - destVideoTransportAddress_IP
  - destVideoTransportAddress_Port
- Adds user login fields (callingPartyLoginUserID and finalCalledPartyLoginUserID) to ensure that the system associates a valid UserID with a newly created phone device and that it gets properly reported in CDRs
- Adds examples for different call scenarios including IDivert, Barge, and cBarge

Cisco CallManager Release 3.2

The change for CDRs in Cisco CallManager release 3.2 means that CDR records get written to comma-delimited flat files (text) on the subscriber databases. The localCDRPath service parameter specifies the directory to which the files are written. CDR and CMR records for each subscriber periodically pass to the publisher database, and the CiscoInsertCDR service reads the records from the flat files and inserts the records into the centralized SQL database.

Cisco CallManager CDR Overview

The Cisco CallManager comprises several Windows 2000 Servers that are using Microsoft SQL clustering method to share common data. Each cluster comprises a publisher and several subscriber databases.

Microsoft SQL Server 2000 Service Pack 3A, which replaces Microsoft SQL 7.0, gets configured with only TCP for communications and NT authentication. Named Pipes communications and Mixed mode authentication no longer get configured.

Note Windows NT Authentication is recommended, although the system supports SQL Authentication. Setting Cisco CallManager for mixed mode authentication in Release 4.0 and later is unsupported. Upgrades will fail and the system will have to be changed back to Windows authentication.

The connection logic in the database layer uses Windows NT authentication. All database layer connections, which are DSN based, use an Open Database Connectivity (ODBC) system DSN, **CiscoCallManager**. For more information, see the "Reading Records" section.

Any third party application that connects to the database can change the way that it connects. Beyond that, everything else remains the same. Both previous and current connections work.

The configuration ensures that web applications that do not require an NT login and use the database layer, such as CCMUser, run as a different NT user with limited privileges, not ANONYMOUS.

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

The CDR records relate to the CMR records via the two globalCallID columns:

- globalCallID_callManagerId
- globalCallID_callId

The database server (publisher) maintains the central copy of the CDR database. When a call is generated on a subscriber, the Cisco CallManager writes CDRs and CMRs in flat files (text) on the subscriber databases. The localCDRPath enterprise parameter specifies the directory to which the files are written. CDR and CMR records periodically pass from each of the subscribers to the publisher, and the Cisco CDR Insert service reads the records from the flat files and inserts the records into the centralized SQL database.

The configurable directory that contains the files defaults to \Program Files\Cisco\CallDetail.

Cisco CallManager does not perform any post processing on the records. For more information, see the "Writing Records" section.

Note Each server (publisher and subscriber) can operate as a call-control engine, but Cisco recommends that you reserve the publisher server for management processes.

Cisco CallManager Configuration

Enable or disable CDRs and CMRs through the Cisco CallManager service parameters. You can find information on where and how the CDRs and CMRs are generated in the System enterprise parameters.

Service Parameters

Cisco CallManager contains the following service parameters, set to False by default, that control the generation of CDRs and CMRs:

- CDR Enabled Flag—Enables or disables CDRs. Set this parameter to True to generate CDRs
- CDR Log Calls with Zero Duration Flag—Enables logging of CDRs for calls that were never connected or that lasted less than 1 second. If you set this parameter to True, all calls get written to the database.
- Call Diagnostics Enabled—Enables or disables CMRs.

To view all CDRs for billing and fraud detection purposes, enable the flags.

The Max CDR Records service parameter for the Cisco Database Layer Monitor service allows you to set a limit on the maximum number of CDRs and CMRs on the system.

Once a day, Cisco CallManager checks the number of CDRs and CMRs on the system and purges CDRs and associated CMRs when the number of CDRs exceeds the value that is specified in the Max CDR Records parameter. At the same time, Cisco CallManager purges orphaned CMRs when the number of CMRs exceeds three times the value that is specified in the Max CDR Records. For example, when you set the Max CDR Records parameter to 1.5 million (default value), Cisco Database Layer Monitor service will purge CDRs and the associated CMRs when the number of CDRs reaches 1.5 million. When orphaned CMRs reach 4.5 million records (three times the 1.5 million default value), Cisco Database Layer Monitor service will purge CMRs. The Cisco Database Layer Monitor service orders the CDRs and CMRs based on their timestamp, determines the timestamp of the record that exceeded the limit, and purges all records with timestamps equal to or earlier than that timestamp.

Warning **Cisco CallManager purges CDRs solely based on the timestamp and will remove CDRs not in that have not been downloaded to a billing system if the CDRs have a timestamp earlier than the time range that has been targeted for purging.**

Note To generate CDRs for all Cisco CallManager servers, enable CDR configurations separately on each server in a cluster by setting the CDR Enabled Flag parameter to True. To generate CMR for all Cisco CallManager servers in the cluster, set the Call Diagnostic Enabled parameter to true on one server.

You can configure service parameters on the Service Parameters Configuration window in the Cisco CallManager Administration. Refer to *Cisco CallManager Administration Guide*.

Enterprise Parameters

Configure the following parameters in the Enterprise Parameters Configuration window in Cisco CallManager Administration.

- Local CDR Path—The directory for local CDR files that Cisco CallManager writes. Ensure that the value is not empty or invalid, or the CDR files will not be moved.
- Primary CDR UNC Path—The central collection point for CDR files. Ensure that the value is not empty or invalid, or the CDR files will not be moved. The install sets this parameter.
- CDR Format—The parameter that determines whether the files get inserted into the database. The value specifies either FLAT or DB(Default DB).
- Primary CDR DSN—An optional parameter that points to the primary CDR server on which to insert CDRs. The machine, to which the parameter points, does not need a Cisco CallManager install but does need SQL server and a CDR database. This allows movement of the CDRs off the Cisco CallManager cluster. If this parameter is missing, CDRs get written locally at the PrimaryCDRUNCPath.
- CDR Flat File Interval—The parameter that determines the number of minutes to write to a CDR file before Cisco CallManager closes the CDR file and opens a new one.

Note If the Primary CDR DSN parameter is missing, CDRs get written locally at the Primary CDR UNC Path.

Retaining the default values for these parameters will write CDRs to the primary CDR server database.

Global Call Identifier

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

The CDR table lists the CDRs that are written to the CDR table 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 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.

The following table contains a sample CDR:

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 GlobalCallID 4 ended, and only completed calls and failed calls get written to the CDR table.

Number Translations

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

For, 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 may 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#.”

The CDR uses the following Partition/Extension Number:

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.

Phone Number	Description
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 CallManager supports multiple Cisco IP Phones with the same extension number in different partitions. For calls that ingress through a gateway, this field remains blank.
originalCalledPartyNumberPartition	This number identifies the partition name that is associated with the OriginalCalledPartyNumber field. This field uniquely identifies this number because the Cisco CallManager supports multiple Cisco 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 CallManager supports multiple Cisco 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 CallManager supports multiple Cisco 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 record appear in universal coordinated time (UTC), which is the number of seconds since midnight on January 1, 1970. 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 Windows NT (2000) system routines.

The CDR includes the following timestamps:

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

Call Clearing Causes

The CDR record includes two clearing causes: OrigCause and DestCause. When the originating party clears the call, the OrigCause gets populated. When the terminating party clears the call, or the call is rejected, the DestCause gets populated. When unpopulated, the cause value shows zero.

The "Cause Codes" section lists the calls clearing cause values per ITU specification Q.850. For on-net call legs, the Cisco CallManager determines the cause value. For off-net call legs, the far-end switch determines the cause value.

IP Addresses

The system stores IP addresses as unsigned integers. The database displays them 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 notation.

Note The database 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:

Procedure

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

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

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

Step 4 The IP address displays in the following format:
192.168.18.188

Working with CDRs

Users can access the Microsoft SQL Server 2000 Service Pack 2 database via ODBC. The install configures an ODBC system DSN that is called **`CiscoCallManager`**. Users have read-only access to all tables in the database and have read/write access to the CDR and CMR tables.

When working with CDRs, you may want to read other tables in the database to obtain information about the type of device in each CDR. Because this correlation between devices in the Device table and the IP address that is listed in the CDR is not straightforward, it appears as a known issue in the "Known Issues" section.

Writing Records

The Cisco CallManager writes CDRs to the SQL database, as calls are made, in a manner consistent with the configuration of each individual Cisco CallManager. You can configure the Cisco CallManager by accessing Service Parameters Configuration in Cisco CallManager Administration.

When CDR records are enabled, Call Control generates one or more CDR records for each call. These records get sent to EnvProcessCdr, where they are written to the flat files. The number of records that are written varies by the type of call and significant changes that occur to the call, such as ending a call, transferring the call, redirecting the call, splitting, or joining a call.

When the Call Diagnostics service parameter is set to true, processStationCdpc generates up to two CMR records 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 the database at the end of the call. Only completed calls and failed calls generate records.

Note The Cisco CDR Insert service will not insert a record if the CDRFormat enterprise parameter has a value of Flat. If the service is disabled on the local Cisco CallManager, the CDR files generate but do not get moved and deleted.

Reading Records

The easiest way to read data from the SQL database may be to use ODBC. The connection string looks like one of the following examples, depending on whether you need to get to the configuration data or CDRs:

For SQL authentication:

DRIVER={SQL Server};SERVER=machineX;DATABASE=CCM0300;UID=CiscoCCMUser;PWD=password

DRIVER={SQL Server};SERVER=machineX;DATABASE=CDR;UID=CiscoCCMCDR;PWD=password

For Windows NT authentication:

DSN=CiscoCallManager;SERVER=X;DATABASE=CCM0300;Trusted_Connection=yes

or

DSN=CiscoCallManager;SERVER=X;DATABASE=CDR;Trusted_Connection=yes

Use the correct database name. The tables reside in the CDR database.

Note You need access to both the configuration database and CDR database to properly resolve the CDR information.

The machine that serves the primary CCM0300 database serves as the machine that is the central collector of the CDR information.

You can find the primary database (machine and name) that the cluster currently is using by opening Cisco CallManager Administration, choosing **Help > About Cisco CallManager**, and clicking the Details button. You can also check the registry on machines that host a database. Look at the registry key, \\HKEY_LOCAL_MACHINE\Software\Cisco Systems Inc.\DBL, for DBConnection0. This string item contains a connection string that includes the machine name and database name of the primary database.

The following table specifies the user ID and password that you should use when you access the Cisco CallManager database.

Database	Tables	SQL User ID	Password	Capability
CDR	CallDetailRecord, CallDetailRecordDiagnostic	CiscoCCMCDR	dipsy	Read/Write
CCM0300	All	CiscoCCMCDR	dipsy	Read only

Removing Records

Because the Cisco CallManager relies on third-party packages to process the CDR data, you should remove the CDR data after all packages finish with the data. Use the CiscoCCMCDR user to remove the records. The CiscoCCMCDR user designates the Microsoft SQL Server account that can be used to read/write to the CDR and CMR tables.

If CDRs accumulate to a configured maximum, the system removes the oldest CDRs along with related CMR records once a day. The default maximum specifies 1,500,000 CDRs.

When removing CDR data after analysis, be sure to remove all related CMR records also.

Tips You should remove records more often than once a day or week in large systems. Queries to remove records consume CPU time and transaction log space relative to the size of the table: the smaller the table, the quicker your query.

CDR Record Field Descriptions

The following table defines all fields in the current CDR records.

Field Name	Range of Values	Description
cdrRecordType	0, 1, or 2	Defines the type of record. The following valid values apply: • 0—Start call detail record (not used) • 1—End call detail record Default - For CDR records, this field will always be 1.
globalCallID_callManagerId	Positive Integer	Designates unique Cisco CallManager identity. This field comprises 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 - This field should always be populated.
globalCallID_callID	Positive Integer	Designates unique call identity value that is assigned to each call. Cisco CallManager allocates this identifier independently on each call server. Values get chosen sequentially when a call begins. A value assignment occurs for each call, successful or unsuccessful. This field comprises half of 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 - This field should always be populated.

Field Name	Range of Values	Description
origLegCallIdentifier	Positive Integer	Identifies the originating leg of a call with a value that is unique within a cluster. 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 - This field should always be populated.
dateTimeOrigination	Integer	Identifies the date and time when the user goes off hook or the date and time when the setup message is received for an incoming call. UTC specifies the time. Default - This field should always be populated.
origNodeId	Positive Integer	Identifies the node within a cluster to which the originator of the call is registered at the time the call is made. Default - This field should always be populated.
origSpan	Positive Integer or Zero	For calls that originate at a gateway, identifies the port or span on the gateway where the call originated. For gateways in which the span number is unknown, this field contains the call leg ID of the originator. For calls that did not originate at a gateway, the value equals zero. Default - Populated based on these rules.
origIpAddr	Integer	Identifies the IP address of the device that originated the call signaling. For Cisco IP Phones, this field specifies the address of the Cisco IP Phone. For PSTN calls, this field specifies the address of the gateway. For intercluster calls, this field specifies the address of the remote Cisco CallManager. The "IP Addresses" section describes the IP address format. Default - Populated based on these rules.

Field Name	Range of Values	Description
origIpPort	Positive Integer	Identifies the IP port number that is associated with the OrigIpAddr field. Cisco CallManager does not use or populate this field. Default - Field will always be 0 or null.
callingPartyNumber	Text String	Specifies numeric string of up to 25 characters. For calls that originate at a Cisco IP Phone, this field shows the extension number of the line that is used. For incoming 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 were applied to the Calling Party Number before it arrives at the Cisco CallManager (such as translations at the gateway). Default - Populated based on these rules.
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 "Cause Codes" section lists the valid values per Q.850. For clearing causes that are created internally by the Cisco CallManager, this value equals zero. Default - 0.

Field Name	Range of Values	Description
origCause_value	0 to 128	For calls that the originating party cleared, reflects the reason for the clearance. The "Cause Codes" section lists the valid values per Q.850. For calls that the terminating party cleared, this field specifies zero. In addition to the standard values that are described in Q.850, when a call is placed on hold, the CDR terminates, and this field gets set to 126. This represents a proprietary value for this field. MLPP uses the current cause codes: • 8—"Preemption: Call is being preempted, circuit not reserved for reuse." • 9—"Preemption: Call is being preempted, circuit reserved for reuse." • 46—"Precedence call blocked: No preemptable circuit or called user is busy with a call of equal or higher precedence level." • 50—"Requested facility not subscribed." Cisco CallManager never generates this cause value. If is is received from another network, the system "transits" this value, if applicable. Default - 0.
origPrecedenceLevel	0 to 4	For MLPP, each call leg has a precedence level. This field represents the original legs precedence level. • Precedence 0 = FLASH OVERRIDE • Precedence 1 = FLASH • Precedence 2 = IMMEDIATE • Precedence 3 = PRIORITY • Precedence 4 = ROUTINE Default - 4.

Field Name	Range of Values	Description
origMediaTransportAddress_IP	Integer	Identifies the IP address of the device that originated the media for the call. For Cisco IP Phones, this field specifies the address of the Cisco IP Phone. For PSTN calls, this field specifies the address of the gateway. For intercluster calls, this field specifies the address of the remote Cisco IP Phone. The "IP Addresses" section describes the IP address format. Default - 0. If media is not established, this field will be 0.
origMediaTransportAddress_Port	Positive Integer	Identifies the IP port number that is associated with the OrigMediaTransportAddress_IP field Default - 0. If media is not established, this field will be 0.

Field Name	Range of Values	Description
origMediaCap_payloadCapability	0 to 15, 32 to 33, 80 to 84	Identifies the codec type that the originator used to transmit media. The following valid values descriptions apply: • 0—Media transfer stage was not reached during the call. • 1—Nonstandard Codec • 2—G.711 A-Law 64K • 3—G.711 A-Law 56K • 4—G.711 mu-Law 64K • 5—G.711 mu-Law 56K • 6—G.722 64K • 7—G.722 56K • 8—G.722 48K • 9—G.723.1 • 10—G.728 • 11—G.729 • 12—G.729AnnexA • 13—Is11172AudioCap • 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 • 80—GSM • 81—ActiveVoice • 82—G726_32K • 83—G726_24K • 84—G726_16K Default - 0. If media is not established, this field will be 0.

Field Name	Range of Values	Description
origMediaCap_maxFramesPerPacket	Positive Integer or Zero	Identifies the number of milliseconds of data per packet that the originating party sent. This field, normally set to 10, 20, or 30 for G.729 or G.711 codecs, can store any nonzero value. Default - 0. If media is not established, this field will be 0.
origMediaCap_g723BitRate	0, 1, or 2	When the codec that the originating party used is G.723, indicates the data rate that was used. The following values apply: • 1—5.3K • 2—6.3K When the codec is not G.723, this value equals zero. Default - Field will always be 0.
origVideoCap_Codec	100 = H.261 101 = H.263 102 = Cisco Video Link	Identifies the codec type used by the originator to transmit video (H.261, H.263, and the Cisco Video Link.) Default - 0. If media is not established, this field will be 0.
origVideoCap_Bandwidth	Positive Integer	Identifies the bandwidth measured in units of kbps. Default - 0. If media is not established, this field will be 0.
origVideoCap_Resolution	1 = SQCIF 2 = QCIF 3 = CIF 4 = CIF4 5 = CIF16	Identifies the Video resolution. Default - 0. If media is not established, this field will be 0.
origVideoTransportAddress_IP	Integer	Identifies the IP address of the device that originates the call. Default - 0. If media is not established, this field will be 0.
origVideoTransportAddress_Port	Positive Integer	Identifies the video RTP port associated with the origVideoTransportAddress_IP field. Default - 0. If media is not established, this field will be 0.

Field Name	Range of Values	Description
destLegIdentifier	Positive Integer	Identifies the terminating leg of a call. This field specifies unique values within a cluster. 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 will be 0.
destNodeId	Positive Integer	Identifies the node within a cluster to which 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 will be 0.
destSpan	Positive Integer or Zero	For calls that terminate at a gateway, identifies the port or span on the gateway where the call terminated. For H.323 gateways in which the span number is unknown, this field contains the call leg ID of the destination. For calls that did not terminate at a gateway, the value equals zero. Default - 0. If the destination cannot be reached, this field will be 0.
destIpAddr	Integer	Identifies the IP address of the device that terminated the call signaling. For Cisco IP Phones, this field specifies the address of the Cisco IP Phone. For PSTN calls, this field specifies the address of the gateway. For intercluster calls, this field specifies the address of the remote Cisco CallManager. The "IP Addresses" section describes the IP address format. Default - 0. If the destination cannot be reached, this field will be 0.
destIpPort	Positive Integer or Zero	Identifies the IP port number that is associated with the destIpAddr field. This field is not used or populated by Cisco CallManager. Default - Field will always be 0 or null.

Field Name	Range of Values	Description
originalCalledPartyNumber	Text String	Specifies numeric string of up to 25 characters. This field specifies the number to which the original call was presented, prior to any call forwarding. If translation rules are configured on the Cisco CallManager, this number reflects the called number after the translations have been applied. Default - empty string “” or null. If destination cannot be reached, this field will be empty.
finalCalledPartyNumber	Text String	Specifies numeric string of up to 25 characters. This field specifies the number to which the call is finally presented, until it is answered or rings out. If no forwarding occurred, this number shows the same number as the 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”). Default - empty string “” or null. If destination cannot be reached, this field will be empty.
destCause_location	0 to 15	For clearing causes that are received over ISDN signaling links, specifies the Location field that the ISDN release message indicates. The “Cause Codes” section lists the valid values per Q.850. For clearing causes that the Cisco CallManager created internally, this value equals zero. Default - 0. If the destination cannot be reached, this field will be 0.

Field Name	Range of Values	Description
destCause_value	0 to 128	For calls that the destination party cleared, reflects the reason for the clearance. The "Cause Codes" section lists the valid values per Q.850. For calls that the originating party cleared, this field equals zero. MLPP uses the current cause codes: • 8—"Preemption: Call is being preempted, circuit not reserved for reuse." • 9—"Preemption: Call is being preempted, circuit reserved for reuse." • 46—"Precedence call blocked: No preemptable circuit or called user is busy with a call of equal or higher precedence level." • 50—"Requested facility not subscribed." Cisco CallManager never generates this cause value. If it is received from another network, the system "transmits" this value, if applicable. • 128—"Conference Drop Any Party." This cause code indicates when a call was dropped from a conference by the new feature "drop any party/drop last party." Default - 0. If the destination cannot be reached, this field will be 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

Field Name	Range of Values	Description
destMediaTransportAddress_IP	Integer	Identifies the IP address of the device that terminated the media for the call. For Cisco IP Phones, this field designates the address of the Cisco IP Phone. For PSTN calls, this field designates the address of the H.323 gateway. For intercluster calls, this field shows the address of the remote Cisco IP Phone. The "IP Addresses" section describes the IP address format. Default - 0. If the destination cannot be reached, this field will be 0.
destMediaTransportAddress_Port	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 will be 0.
destMediaCap_payloadCapability	0 to 15, 32 to 33, 80 to 84	Identifies the codec type that the terminating party used to transmit media. The Codec Types section lists the valid values. Default - 0. If the destination cannot be reached, this field will be 0.
destMediaCap_maxFramesPerPacket	Positive Integer	Identifies the number of milliseconds of data per packet that the terminating party of the call sent. This field, normally set to 10, 20, or 30 for G.729 or G.711 codecs, can store any nonzero value. This field can be zero if the media is never established. Default - 0. If the destination cannot be reached, this field will be 0.
destMediaCap_g723BitRate	0	Depreciated since Cisco CallManager Release 3.3(4). Default - This field is always 0.
destVideoCap_Codec	100 = H.261 101 = H.263 102 = Cisco Video Link	Identifies the codec type that the terminating party used to transmit video (H.261, H.263, and the Cisco Video Link). Default - 0. If the destination cannot be reached, this field will be 0.

Field Name	Range of Values	Description
destVideoCap_Bandwidth	Positive Integer	Identifies the bandwidth measured in units of kbps. Default - 0. If the destination cannot be reached, this field will be 0.
destVideoCap_Resolution	1 = SQCIF 2 = QCIF 3 = CIF 4 = CIF4 5 = CIF16	Identifies the video resolution. Default - 0. If the destination cannot be reached, this field will be 0.
destVideoTransportAddress _IP	Integer	Identifies the IP address of the device that receives the call. Default - 0. If the destination cannot be reached, this field will be 0.
destVideoTransportAddress_Port	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 will be 0.
dateTimeConnect	Integer or Zero	Identifies the date and time that the call connected. UTC specifies the time. If the call is never answered, this value shows zero. Default - 0. If the call is never connected, this field will be 0.
dateTimeDisconnect	Integer	Identifies the date and time when the call was cleared. This field gets set even if the call never connected. UTC specifies the time. Default - 0.

Field Name	Range of Values	Description
lastRedirectDn	Text String	Specifies numeric string of up to 25 characters. 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 “” or null. If call was never redirected, this field will be 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	Identifies the partition name associated with the OriginalCalledPartyNumber field. This field uniquely identifies this number because the Cisco CallManager supports multiple Cisco 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. Default - empty string “”or null. If the original called party does not have a partition, this field will be empty.

Field Name	Range of Values	Description
callingPartyNumberPartition	Text String	Identifies the partition name that is associated with the CallingPartyNumber field. This field uniquely identifies this number because the Cisco CallManager supports multiple Cisco IP Phones with the same extension number in different partitions. For calls that ingress through a gateway, this field remains blank. Default - empty string “” or null. If the original called party does not have a partition, this field will be empty.
finalCalledPartyNumberPartition	Text String	Identifies the partition name that is associated with the FinalCalledPartyNumber field. This field uniquely identifies this number because the Cisco CallManager supports multiple Cisco 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 pointed to the gateway. Default - empty string “” or null. If the final called party does not have a partition, this field will be empty.
lastRedirectDnPartition	Text String	Identifies the partition name that is associated with the LastRedirectDn field. This field uniquely identifies this number because the Cisco CallManager supports multiple Cisco 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 pointed to the gateway. Default - empty string “” or null. If the last redirecting Party does not have a partition or the call was never redirected, this field will be empty.
duration	Positive Integer or Zero	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 connected or connected for less than 1 second. Default - 0.

Field Name	Range of Values	Description
origDeviceName	Text String	Specifies text string that identifies the name of the originating device. Default - This field will always be populated.
destDeviceName	Text String	Specifies text string that identifies the name of the destination device. Default - empty string "" or null. If the original device does not have a name, this field will be empty.
origCalledPartyRedirectReason	Integer	Identifies the reason for a redirect of the original called party. See the "Redirect Reason Codes" section for a complete list of the codes. Default - 0.
lastRedirectRedirectReason	Integer	Identifies the last redirect reason for redirection. See the "Redirect Reason Codes" section for a complete list of the codes. Default - 0.
destConversationID	Integer	Specifies unique identifier that is used to identify the parties of a conference call. Default - 0.
origCallTerminationOnBehalfOf	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 is terminated because of a transfer, the OnBehalfOf code shows "10." See the "OnBehalfof Codes" section for a complete list of the codes. Default - 0.
destCallTerminationOnBehalfOf	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 is terminated because of a transfer, the OnBehalfOf code shows "10." See the "OnBehalfof Codes" section for a complete list of the codes. Default - 0.

Field Name	Range of Values	Description
origCalledPartyRedirectOnBehalfOf	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 for a complete list of the codes. Default - 0.
lastRedirectRedirectOnBehalfOf	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 for a complete list of the codes. Default - 0.
joinOnBehalfOf	Integer	Specifies code that identifies the reason for a join. For example, if the join took place on behalf of a transfer, the OnBehalfOf code specifies “10.” See the “OnBehalfof Codes” section for a complete list of the codes. Default - 0.
globalCallId_ClusterId	Text String	Specifies a unique ID that identifies a cluster of Cisco CallManagers. Cisco CallManager does not use this field that generates at installation. The following fields make up this unique key: GlobalCallId_ClusterId + GlobalCallId_CMId + GlobalCallId_CallId Default - This field should always be populated.
Comment	TextString	This field allows features to add text to the CDR records. This text can describe details about the call. For example, the following field flags malicious calls. Tag—CallFlag Value—MALICIOUS Default - This empty string “” or null.

Field Name	Range of Values	Description
callingPartyLoginUserID	Text String	Identifies the calling party user login ID. Default: This empty string, "", or null.
finalCalledPartyLoginUserID	Text String	Identifies the final called party user login ID. Default: This empty string, "", or null.
authCodeDescription	Text String	Description of the authorization code. For security purposes, the authorization code does not get written to the CDR; the authorization description and level get written instead. Default: This empty string, "", or null.
authorizationLevel	0, integer	The level of the authorization code. For security purposes, the authorization does not get written to the CDR; the authorization description and level get written instead. 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. Default: This empty string, "", or null. "9999999999999999" indicates an invalid CMC in CDR for a zero duration call.

CMR Fields (Diagnostic)

The following table contains the fields, range of values, and field descriptions of the CMRs.

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 CMR records, this field will always be 2.
globalCallID_callManagerId	Positive Integer	Specifies a unique Cisco CallManager 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 - This field should always be populated.
globalCallID_callId	Positive Integer	Specifies a unique call identity value that is assigned to each call. This identifier gets allocated independently on each call server. Cisco CallManager chooses values 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 - This field should always be populated.

Field Name	Range of Values	Description
nodeId	Positive Integer	Specifies the node within the Cisco CallManager cluster where this record generates. Default - This field should always be populated.
callIdentifier	Positive Integer	Specifies a call leg identifier that identifies the call leg to which this record pertains. Default - This field should always be populated.
directoryNumber	Integer	Specifies the directory number of the device from which these diagnostics were collected. Default - This field should always be populated.
dateTimeStamp	Integer	Represents the approximate time that the device went on hook. Cisco CallManager records the time when the phone responds to a request for diagnostic information. Default - This field should always be populated.
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 were 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.

Field Name	Range of Values	Description
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 were 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.
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. 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.

Field Name	Range of Values	Description
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 CallManager obtains the average by summing all estimates then dividing by the number of RTCP messages that have been received. Default - 0.
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 - This field will always be populated with a unique id.
directoryNumberPartition	Text String	Identifies the partition of the directory number. Default - This empty string, “”, or null. This field may be empty if no partition exists.
deviceName	Text String	Identifies the name of the device. Default - empty “”or null. This field may be empty if no partition exists.

Field Name	Range of Values	Description
globalCallId_ClusterId	Text String	Designates unique ID that identifies a cluster of Cisco CallManagers. Cisco CallManager does not use this field that is generated during installation: globalCallId_ClusterId + globalCallId_CMId + globalCallId_CallId. Default - This field will always be populated.
varVQMetrics	Text String	This field contains a variable number of voice quality metrics. In this release, the kfactor and concealed seconds VQ (voice quality) metrics are added. This field is a string of voice quality metrics separated by a semicolon. The format of the string is: fieldName=value;fieldName=value/precision This is an example of voice quality data, but the names may be different. "MLQKav=14396/4096;MLQK=13926/4096;MLQKmn=12288/4096; MLQKmx=14396/4096;CRRav=0/65536; CRR=1024/65536; CRRmn=0/65536;CRRmx=3277/65536;CS=15;SCS=2" **Note** See Table 1 "K-Factor Data Stored in Cisco CallManager CMRs" for a complete list of K-Factor data.

K-Factor Data in CMRs

K-factor is an endpoint mean opinion score (MOS) estimation algorithm defined in ITU standard P.VTQ. It is 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 is a term that 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.

The MOS estimate is a number 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 is termed "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 caused by effective packet loss such as dropouts and warbles. It does not estimate the impact of delay-related impairments such as echo. It is 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).

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 eight seconds. Hence, for more accurate scores, k-factor estimates are generated for every eight seconds of active speech.

K-factor and other MOS estimators are considered 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 are primary statistics because they alert the network operator before network impairment has an audible impact, or is visible through MOS.

Table 1 K-Factor Data Stored in Cisco CallManager CMRs

Field Name	Phone Display Name	D&I User Interface Text and Description
CCR	Cum Conceal Ratio	Cumulative Conceal Ratio. Is the cumulative ratio of concealment time over speech time observed after starting a call.
ICR	Interval Conceal Ratio	Interval Conceal Ratio. Is an interval-based average concealment rate, and is the ratio of concealment time over speech time for the last three seconds of active speech.
ICRmx	Max Conceal Ratio	Interval Conceal Ratio Max. Is the maximum concealment ratio observed during the call.
CS	Conceal Secs	Conceal Secs. Is the duration of time during which some concealment is observed during a call.
SCS	Severely Conceal Secs	Severely Conceal Secs. Is the duration of time during which a significant amount of concealment is observed. If the concealment observed is usually greater than fifty milliseconds or approximately five percent, the speech is probably not very audible.
MLQK	MOS LQK	MOS Listening Quality K-factor. An estimate of the MOS score of the last eight seconds of speech on the reception signal path.
MLQKmn	Min MOS LQK	MOS Listening Quality K-factor Min. The minimum score observed since the beginning of a call, and represents the worst sounding eight second interval.

Table 1 K-Factor Data Stored in Cisco CallManager CMRs (continued)

Field Name	Phone Display Name	D&I User Interface Text and Description
MLQKmx	Max MOS LQK	MOS Listening Quality K-factor Max. The maximum score observed since the beginning of a call, and represents the best sounding eight second interval.
MLQKav	Avg MOS LQK	MOS Listening Quality K-factor Avg8. The running average of scores observed since the beginning of a call.

Codec Types

The following table contains the compression and payload types that may appear in the codec fields.

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
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
80	GSM
81	ActiveVoice
82	G726_32K

Value	Description
83	G726_24K
84	G726_16K
100	H261
101	H263
102	Cisco Video Link

Cause Codes

The following table contains cause codes that may appear in the Cause fields.

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

Code	Description
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	Bearer capability not presently available
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 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)
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

Code	Description
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 a timer expired, and 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 (this is a Cisco-specific code)
123	Device Not Preempt able (this is a Cisco-specific code)
124	Conference Full (this is a Cisco-specific code)
125	Out of bandwidth (this is a Cisco-specific code)
126	Call split. This Cisco-specific 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 designation can help determine which calls terminated as part of a transfer operation.
127	Interworking, unspecified
128	Drop any party/drop last party conference feature
129	Precedence out of bandwidth

Redirect Reason Codes

The following table contains the available Redirect Reason Codes that may appear in a record.

Value	Description
Q.931 Standard	**Redirect Reason Codes**
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

Value	Description
Q.931 Standard	**Redirect Reason Codes**
Non Standard	**Redirect Reason Codes**
16 + 2	Call Deflection
32 + 2	Blind Transfer
48 + 2	Call Immediate Divert
64 + 2	Call Forward Alternate Party
80 + 2	Call Forward On Failure
96 + 2	Conference
112 + 2	Barge

OnBehalfof Codes

The following table contains the available OnBehalfof Codes that may appear in a record.

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
12	Device
13	Call Control
14	Immediate Divert
15	Barge
16	Pickup

Call Types

A successful call between two parties logs one CDR record. Each CDR record contains all previously identified fields, but some fields may not be used. If a field is not used, it will be blank if it is an ASCII string field or "0" if it is a numeric field. When supplementary services are involved in a call, more CDR records may be written.

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

This section describes the records that get written for different call types in the system.

Successful On-Net Calls

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

The following table contains two examples:

- A—A 60-second call terminated by the caller
- B—A 60-second call cleared by the called party

	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 record.
- If the call was 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 the them, the "destIpAddr," and the "dateTimeConnect" fields 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 dialed a Directory Number and abandons the call before it connected, the "First Dest" and "Final Dest" fields and their associated partitions will contain the directory number and partition to which the call would have been extended. The "Dest Ip" field remains blank and the duration specifies 0 second.

The following table contains two examples:

- A—On-net call, destination is engaged.
- B—On-net call, destination rings out.

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

Incoming PSTN Calls

The origDeviceName identifies incoming calls. If origDeviceName matches any Gateway device name, it designates an incoming call. The Calling Party number specifies the number that the gateway delivers.

The following table contains three examples:

- A—Successful incoming PSTN call, cleared by caller (PSTN phone)
- B—Successful incoming PSTN call, cleared by called party (Cisco IP Phone)
- C—Call from PSTN to an invalid Cisco IP Phone extension

	Calling Party	Calling Partition	Original Called Party	Original Called Partition	Orig Cause	Dest Cause	Duration
A	02920262227		2309	Marketing	16	0	60
B	02920262227		2309	Marketing	0	16	60
C	02920262227				1	0	0

Outgoing PSTN Calls

You can distinguish outgoing PSTN calls either by the partition name or by the Dialed Number (which begins “9”). These examples use “PSTN” as the partition name. Several partition names may represent the PSTN to achieve a varying class of service.

The following table contains these examples:

- A—Successful outgoing PSTN call, cleared by caller (Cisco IP Phone)
- B—Successful outgoing PSTN call, cleared by called party (PSTN phone)
- C—Successful call to premium rate number
- D—Successful call to premium rate number. Caller uses a # to speed up dialing. (The # key indicates to the Cisco CallManager that all digits have been entered.)
- E—Successful call to mobile number
- F—Successful call to operator

	Calling Party	Calling Partition	Original Called Party	Original Called Partition	Orig Cause	Dest Cause	Duration
A	2001	Accounts	902920262226	PSTN	16	0	60
B	2001	Accounts	902920262226	PSTN	0	16	60
C	2001	Accounts	90891005100	PSTN	0	16	60
D	2001	Accounts	90891005100#	PSTN	0	16	60
E	2001	Accounts	907808784185	PSTN	0	16	60
F	2001	Accounts	9100	PSTN	0	16	60

Call Failures

The system logs all failed outgoing calls whether they have a CdrLogCallsWithZeroDurationFlag set at **True** or **False**, a duration of zero, and a DateTimeConnect value of zero. A failed call can represent anything from a Cisco IP Phone going off hook then immediately on hook to a call to an invalid number.

The following table contains four examples:

- A—Extension 2001 going off hook then on hook (when the CdrLogCallsWithZeroDurationFlag is set at only **True**).
- B—Call to PSTN number, party engaged (cause 17 = user busy).
- C—Call to PSTN number, number does not exist (cause 1 = number unavailable).
- D—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	DateTime Connect	Duration
A	2001	Accounts			16	0	0	0
B	2001	Accounts	902920262226	PSTN	0	17	0	0
C	2001	Accounts	902920100000	PSTN	0	1	0	0
D	2001	Accounts	902920262226	PSTN	0	38	0	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 CDRs. 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.

The following table contains an example of a successful on-net call with a duration of less than 1 second, cleared by the called party.

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

Cisco IP Phone Failure During a Call

When a Cisco IP Phone is unplugged, no immediate, physical indication goes to the Cisco CallManager. The Cisco CallManager relies upon a transmission control protocol (TCP)-based keepalive signaling mechanism to detect when a Cisco IP Phone becomes disconnected.

Each Cisco IP Phone sends a keepalive message to the Cisco CallManager at the configured keepalive interval (default=30 seconds), and the Cisco CallManager responds with an acknowledgement. Both parties then know that the other is functioning properly. When a Cisco IP Phone is unplugged, it fails to send this keepalive message. The Cisco CallManager waits twice the keepalive interval from the time of the last keepalive message before assuming that the Cisco IP Phone no longer functions.

The implication to billing is that, when a Cisco IP Phone is unplugged, the duration of the call that is reflected in the CDR can be up to twice the keepalive interval plus the TCP retry timers longer than the actual speech-time that the user experienced. This worst-case value assumes that the other party did not hang up.

Identify calls that fail in this manner by a cause value of 41 (Temporary Failure). This cause value can possibly occur in other circumstances because external devices such as gateways can also generate this cause value.

The following table contains an example of a successful call from 2001 to 2309 that was terminated by unplugging extension 2001.

Calling Party	Calling Partition	Original Called Party	Original Called Partition	Orig Cause	Dest Cause	Duration
2001	Accounts	2309	Marketing	41	0	120

Pickup Calls

Cisco CallManager includes two pickup modes: Pickup and. This section describes the CDR records for each mode.

Auto Pickup

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

The first CDR get generated 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 will contain 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.

The following steps provide an example of the auto pickup call flow:

Step 1 A call comes in from the PSTN to extensions 2001 and 2002, which are in the same pickup group.

Step 2 Extension 2002 picks up the call that is ringing on 2001.

Step 3 The call automatically connects between the PSTN caller and extension 2002.

The following table shows the CDR content for this call flow:

Call ID	Orig Cause	Calling Party	Dest Cause	Original Called Party	Final Called Party	Last Redirect Party	Orig Term On BehalfOf	Dest Term 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	12	16	16	5	16

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 will 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 will 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 will contain the directory number and partition of the last phone that redirected this call. The redirect on-behalf-of fields will contain 5 (Call Forward) and the redirect reason fields will contain 5 (Pickup). Pickup CDRs look the same for all types of pickup: Pickup, Group Pickup and Other Pickup.

The following steps provide an example of the pickup call flow:

Step 1 A call comes in from the PSTN to extensions 2000, 2001 and 2002, which are in the same pickup group.

Step 2 Extension 2002 picks up the call that is ringing on 2001.

Step 3 Extension 2002 answers the call, and the call connects between the PSTN caller and extension 2002.

The following table shows the CDR content for this call flow:

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

Forwarded or Redirected Calls

Forwarded calls generate a single CDR and show the Calling Party, Originally Called Number, Last Redirecting Number, and Final Called Number. 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 CDR records 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 gets forwarded, the lastRedirectDn and lastRedirectDnPartition fields contain the directory number and partition of the last phone that forwarded or redirected the call.

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

OrigCalledParty Redirected OnBehalfOf	Last Redirect Redirect OnBehalfOf
5	5
5	5

Conference Calls

Two major operational factors exist for Conference 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. In previous releases, the last two parties remained connected to the conference bridge until the call ended.

   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, there 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 conference controller information gets added 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 identifies 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 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.

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 record exists for each party in the conference, one CDR record 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 CDR records exist:

- One CDR record for the original call
- Three CDR records for the parties that are connected to the conference
- One CDR record for the setup/consultation call
- One CDR for the final two parties

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 CallManager. 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 CDR records.

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

- finalCalledPartyNumber—Represents a conference bridge “b0019901001.”
- origCalledPtyRedirectOnBehalfOf—Set to Conference (4).
- lastRedirectRedirectOnBehalfOf—Set to Conference (4).
- joinOnBehalfOf—Set to Conference (4).

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).

The following tables contain these examples:

- Call from 2001 to 2309.
- After 60 seconds, user 2001 presses the "conference" key on the Cisco 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
2001	ACNTS	101	2309	MKTG	102	2309	MKTG	2001
2001	ACNTS	101	2309	MKTG	115	b0029901001		b0029901001
2309	ACNTS	101	b0029901001		116	b0029901001		b0029901001
3071111	PSTN	101	b0029901001		117	b0029901001		b0029901001
2001	ACNTS	105	3071111	PSTN	106	3071111	PSTN	3071111
2001	ACNTS	101	2309	MKTG	102	2309	MKTG	b0029901001

Last Redirect Reason	OrigConversationId	OrigCall TerminationOnBehalfOf	DestCall Termination OnBehalfOf	OriginalCalled PtyRedirectOn BehalfOf	LastRedirect Redirect OnBehalfOf	JoinOnBehalf Of	Duration
0	0	4	4	0	0	0	60
0	1	12	0	4	4	4	360
0	1	12	0	4	4	4	360
0	1	4	4	4	4	4	360
0	0	4	4	0	0	0	20
98	0	12	42	0	4	4	55

Meet-Me Conferences

A Meet-Me conference occurs when several parties individually dial into a conference bridge at a predetermined time. In the following examples, 5001 specifies the dial-in number. The conference bridge device signifies special significance to the Cisco CallManager, and calls to the conference bridge appear as forwarded calls; that is, the user 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."

The following tables contain these examples of a call from 2001, 2002, and 2003 that are dialing into a Meet-Me conference bridge with phone number 5001.

	Calling Party	Calling Partition	Original Called Party	Original Called Partition	Final Called Party	Final Called Partition	Last Redirect Party
A	2001	Accounts	5001		b0019901001		b0019901001
A	2002	Accounts	5001		b0019901001		b0019901001
A	2003	Accounts	5001		b0019901001		b0019901001

	Last Redirect Partition	Duration
A		70
A		65
A		60

Call Hold and Resume

When a Cisco IP Phone places an active call on hold and then returns to the call without making a second call, the CDR reflects the entire duration of the original call as an uninterrupted call.

The following table contains an example of a call from Cisco IP Phone 2001 to Cisco IP Phone 2309, placing the call on hold, and resuming speech part way through the call:

Calling Party	Calling Partition	Original Called Party	Original Called Partition	Final Called Party	Final Called Partition	Last Redirect Party	Last Redirect Partition	Duration
2001	Accounts	2309	MKTG	2309	MKTG	2309	MKTG	70

Transfer Without Consultation

A single CDR cannot show call transfer, which is too complex to show. Each time that a call is transferred, the Cisco CallManager terminates the CDR for that call. 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.

The following table contains three examples:

- A—Call from extension 2001 to a PSTN number, talking for 120 seconds.
- B—Extension 2001 initiates a transfer without consultation (hence the 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
A	2001	ACNTS	101	3071111	PSTN	102	126
B	2001	ACNTS	103	2002	ACNTS	104	126
C	3071111	PSTN	102	2002	ACNTS	104	0

	Dest Cause	OrigCall Termination OnBehalfOf	DestCall Termination OnBehalfOf	Join OnBehalf Of	Duration
A	126	10	10	0	120
B	126	10	10	0	0
C	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 CallManager 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.

The following tables contain three examples:

- A—Call from extension 2001 to a PSTN number, 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
A	2001	ACNTS	101	3071111	PSTN	102	126

	Calling Party	Calling Partition	Calling Leg	Original Called Party	Original Called Partition	Called Leg	Orig Cause
B	2001	ACNTS	103	2002	ACNTS	104	126
C	3071111	PSTN	102	2002	ACNTS	104	0

	Dest Cause	OrigCall Termination OnBehalfOf	DestCall Termination OnBehalfOf	Join OnBehalf Of	Duration
A	126	10	10	0	120
B	126	10	10	0	30
C	16	0	0	10	350

Precedence Calls (MLPP)

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

The following table contains an example CDR for this scenario:

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

Calling Party	Calling Partition	Orig 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 CallManager network flags the call. The Comment field flags the malicious call.

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.

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)

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

The following table contains an example CDR for this scenario:

Calling Party	Calling Partition	Original Called Party	Original Called Partition	Final Called Party	Final Called Partition	Last Redirect Party	Last Redirect Partition	Duration
02920262227		2001	ACNTS	2309	MKTG	2001	ACNTS	120
02920262227		2001	ACNTS	6000	VMAIL	2309	MKTG	60

OrigCalledPartyRedirectedOnBehalfOf	LastRedirectRedirectOnBehalfOf
5	5
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 - 320K
- 2 = QCIF

Calling Party	Calling Partition	Calling Leg	Original Called Party	Original Called Partition	Called Leg	OrigVideo Cap_Codec	OrigVideo Cap_Bandwidth	OrigVideo Cap_Resolution
51234	CISCO	101	57890	CISCO	102	100	320	2

OrigVideo Transport Address_IP	OrigVideo Transport Address_Port	DestVideo Cap_Codec	DestVideo Cap_Bandwidth	DestVideo Cap_Resolution	DestVideo Transport Address_IP	DestVideo Transport Address_Port
187962284	49208	100	320	2	288625580	49254

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 called person's name. 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 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.

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 Number	OrigLegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number
2101	16777217	PAManaged	16777219	2004	Phones	2000
2004	16777221	Phones	16777222	2105	PAManaged	2105
2101	16777217	PAManaged	16777222	2105	PAManaged	2105

Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration (in seconds)
1023970182	2000	Phones	34
1023970182	2105	PAManaged	0
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. See the sections on "Transfer Without Consultation" and "Forwarded or Redirected Calls".

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 Number	OrigLegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number
2002	16777234	Phones	16777285	2105	PAManaged	2105
2101	16777230	PAManaged	16777232	2002	PA	2105
2105	16777235	PAManaged	16777230	2101	" "	" "

Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration (in seconds)
1023970478	2105	PAManaged	2
1023970478	21xx	" "	9
1023970483	" "	" "	5

Personal Assistant Interceptor Going Directly to Destination

This scenario can have two different cases: with no rules and with rules. The following tables contain examples of each case.

Personal Assistant Going Directly to Destination with No Rules

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	OrigLegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number
2101	16777240	PAManaged	16777242	2105	PA	2105

Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration (in seconds)
1023970710	21XX	" "	8

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

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	OrigLegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number
2101	16777248	PAManaged	16777250	2110	PA	2105

Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration (in seconds
1023970922	21XX	" "	5

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).

The following sections contain examples of each case.

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

The following table contains an example CDR for this scenario:

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

Calling Party Number	OrigLegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number
2004	16777262	Phones	16777263	2110	PAManaged	2110
2101	16777258	PAManaged	16777260	2004	Phones	2000
2110	16777263	PAManaged	16777258	2101	“ “	“ “

Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration (in seconds)
1023971303	2110	PAManaged	6
1023971303	2000	Phones	22
1023971312	“ “	“ “	9

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

The following table contains an example CDR for this scenario:

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

Calling Party Number	OrigLegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number
2001	16777269	Phones	16777270	2110	PAManaged	2110
2001	16777272	Phones	16777273	2120	PAManaged	2120
2101	16777265	PAManaged	16777267	2001	Phones	2000
2120	16777273	PAManaged	16777265	2101	" "	" "
2110	16777275	PAManaged	0	" "	" "	" "

Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration
1023971456	2110	PAManaged	0
1023971467	2120	PAManaged	4
1023971467	2000	Phones	37
1023971474	" "	" "	7
1023971476	" "	" "	0

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

The following table contains an example CDR for this scenario:

- 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 Number	OrigLegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number
2002	16777281	Phones	16777282	2110	PAManaged	2110
2002	16777284	Phones	16777285	2120	PAManaged	2120
2101	16777277	PAManaged	16777279	2002	Phones	2000
2002	16777287	Phones	16777288	2105	PAManaged	2105
2101	16777277	PAManaged	16777288	2105	PAManaged	2105
2105	16777289	PAManaged	0	" "	" "	" "

Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration (in seconds)
1023971602	2110	PAManaged	0
1023971615	2120	PAManaged	0
1023971619	2000	Phones	38
1023971619	2105	PAManaged	0
1023971627	2105	PAManaged	7
1023971629	" "	" "	0

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

The following table contains an example CDR for this scenario:

- User A calls Personal Assistant and says "call User B."
- User B answers the call at extension 2110.

Calling Party Number	OrigLegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number
2003	16777295	Phones	16777296	2110	PAManaged	2110
2101	16777291	PAManaged	16777293	2003	PA	2105
2110	16777296	PAManaged	16777291	2101	" "	" "

Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration (in seconds)
1023971740	2110	PAManaged	4
1023971740	21XX	" "	10
1023971749	" "	" "	9

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

The following table contains an example CDR for this scenario:

- User A calls Personal Assistant and says "call User B."
- User B answers the call at extension 2120.

Calling Party Number	OrigLegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number
2004	16777302	Phones	16777303	2110	PAManaged	2110
2004	16777305	Phones	16777306	2120	PAManaged	2120
2101	16777298	PAManaged	16777300	2004	PA	2105
2120	16777306	PAManaged	16777298	2101	" "	" "

Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration (in seconds)
1023971815	2110	PAManaged	0
1023971824	2120	PAManaged	3
1023971824	21XX	" "	22
1023971832	" "	" "	8

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

The following table contains an example CDR for this scenario:

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

Calling Party Number	OrigLegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number
2001	16777312	Phones	16777313	2110	PAManaged	2110
2001	16777315	Phones	16777316	2120	PAManaged	2120
2101	16777308	PAManaged	16777310	2001	PA	2105
2001	16777318	Phones	16777319	2105	PAManaged	2105
2101	16777308	PAManaged	16777319	2105	PAManaged	2105

Original Called Party Number Partition	Last Redirect DN	Last Redirect DN Partition	Duration (in seconds)
1023971923	2110	PAManaged	0
1023971936	2120	PAManaged	0
1023971940	21XX	" "	30
1023971940	2105	PAManaged	0
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.

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 Number	OrigLegCall Identifier	Calling Party Number Partition	DestLeg Identifier	Final Called Party Number	Final Called Party Number Partition	Original Called Party Number
2003	16777345	Phones	16777346	2105	PAManaged	2105
2101	16777340	PAManaged	16777342	2003	Phones	2000
2003	16777350	Phones	16777351	2002	PAManaged	2110
2003	16777342	Phones	16777347	2110	" "	b00110201001
2110	16777351	PAManaged	16777352	b00110201001	" "	b00110201001
2105	16777346	PAManaged	16777349	b00110201001	" "	b00110201001
2101	16777340	PAManaged	16777348	b00110201001	" "	b00110201001

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

Call Scenarios

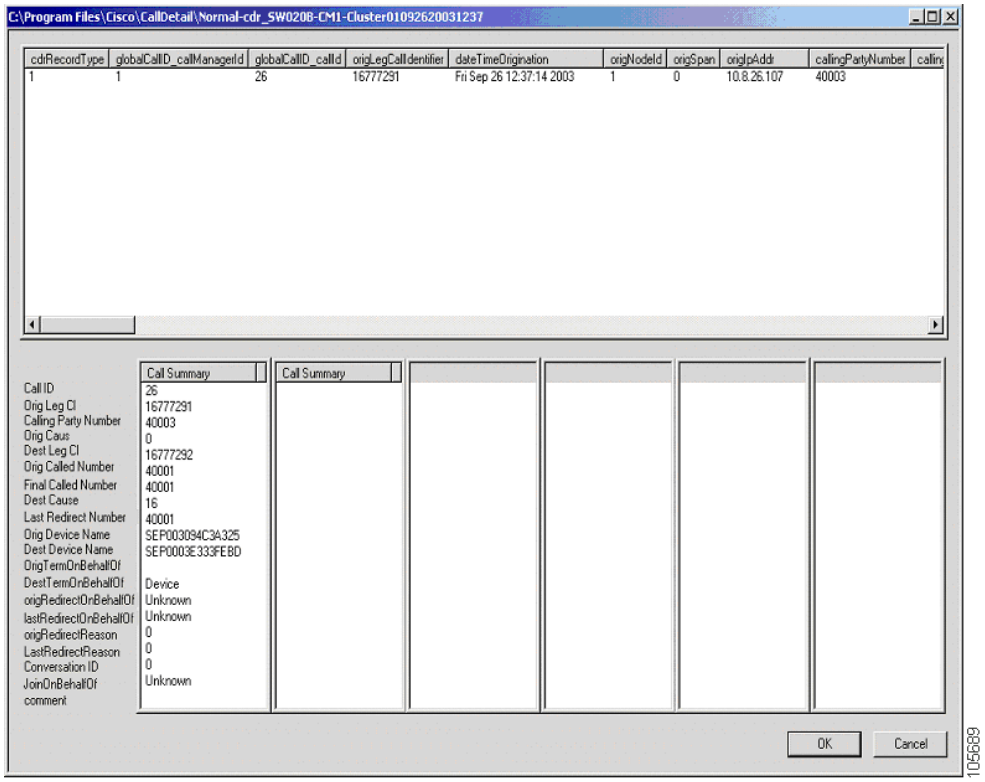
This section displays different call scenarios, including all records for each call and important fields shown in summary screens for easy viewing and comparison.

Normal Call

Figure 1 shows a view of a normal call with the following scenario:

- 40003 calls 40001.
- 40001 answers, talks for 10 seconds, and hangs up.

Figure 1 Normal Call



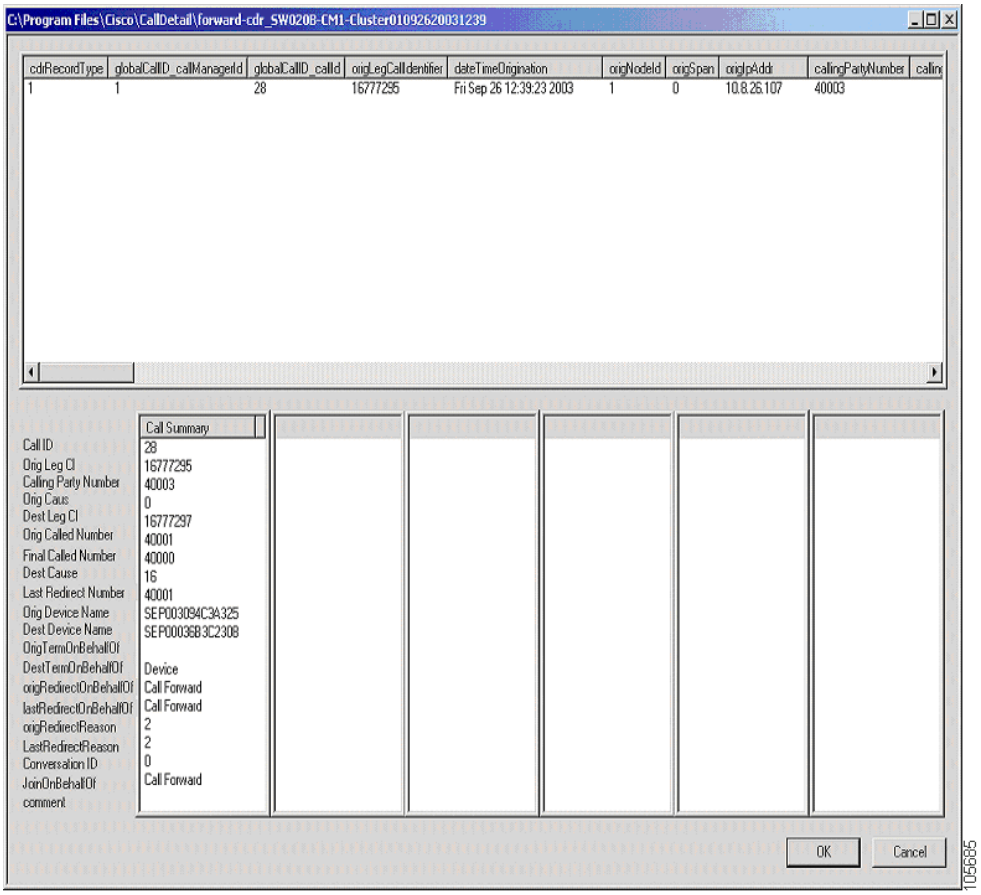
Forwarded Call

Figure 2 shows a view of a forwarded call with the following scenario:

- 40003 calls 40001.
- 40001 CFNA to 40000.
- 40000 answers and hangs up.

Note The original called number equals 40001. The final called number equals 40000. This indicates that the call gets redirected.

Figure 2 Forwarded Call



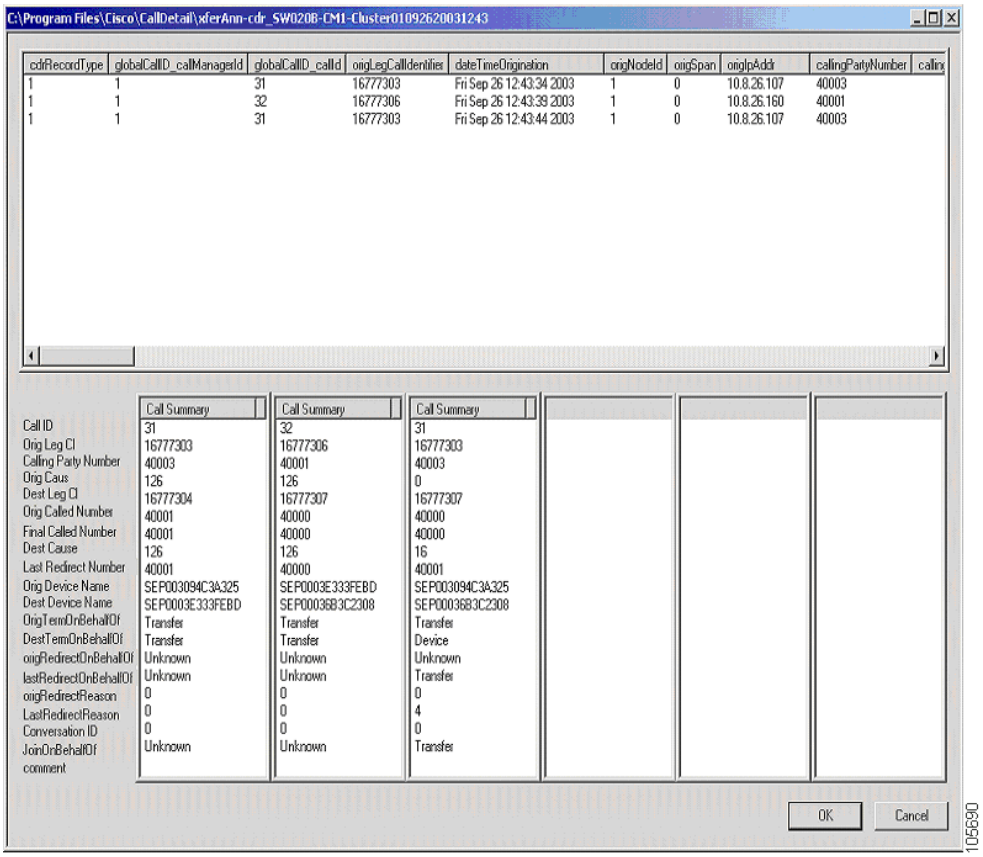
Transfer

Announced Transfer

Figure 3 shows a view of an announced transferred call with the following scenario:

- 40003 calls 40001.
- 40001 presses the transfer button and calls 40000.
- 40000 answers the call.
- 40001 presses the transfer button to complete the transfer.

Figure 3 Announced Transferred Call

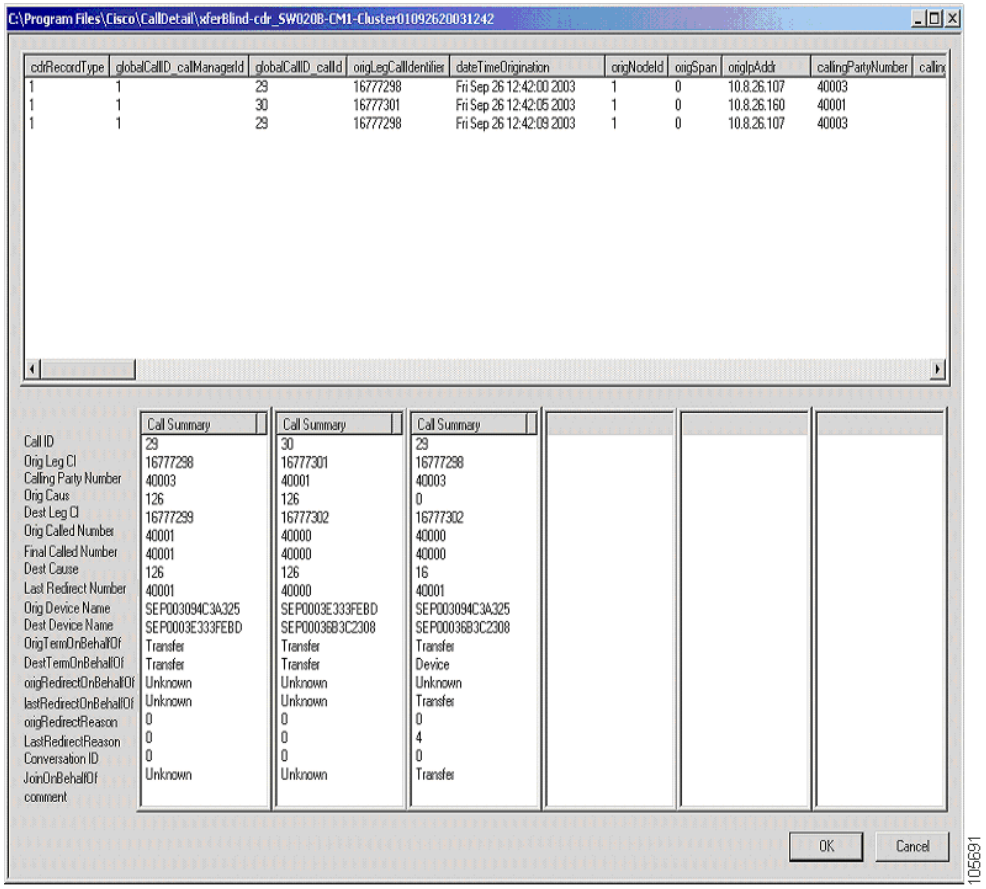


Blind Transfer

Figure 4 shows a view of a blind transfer call with the following scenario:

- 40003 calls 40001.
- 40001 presses the transfer button and calls 40000.
- 40001 presses the transfer button the complete the transfer.

Figure 4 Blind Transfer Call



Ad Hoc Conference

Announced Conference

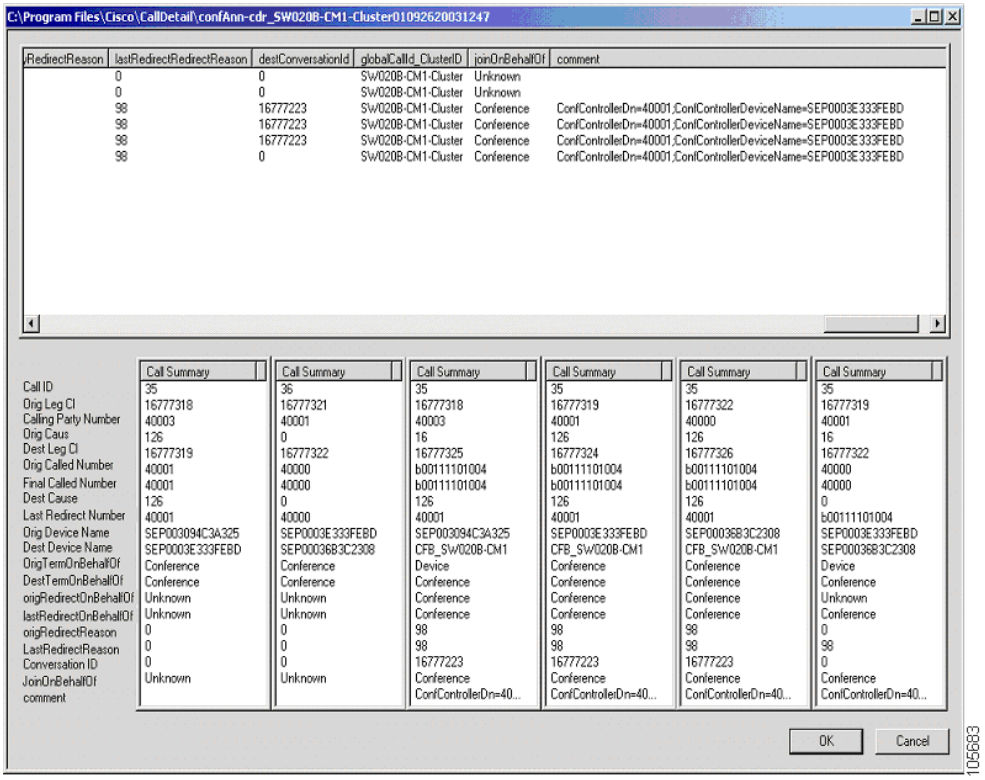
Figure 5 shows a view of an announced conference call with the following scenario:

- 40003 calls 40001.
- 40001 presses the conference button and calls 40000.
- 40000 answers the call.

- 40001 presses the conference button to complete the transfer.
- 40003 hangs up first, and 40000 and 40001 are joined in a direct call (last CDR generated).

Note The comment field identifies the controller.

Figure 5 Announced Conference Call



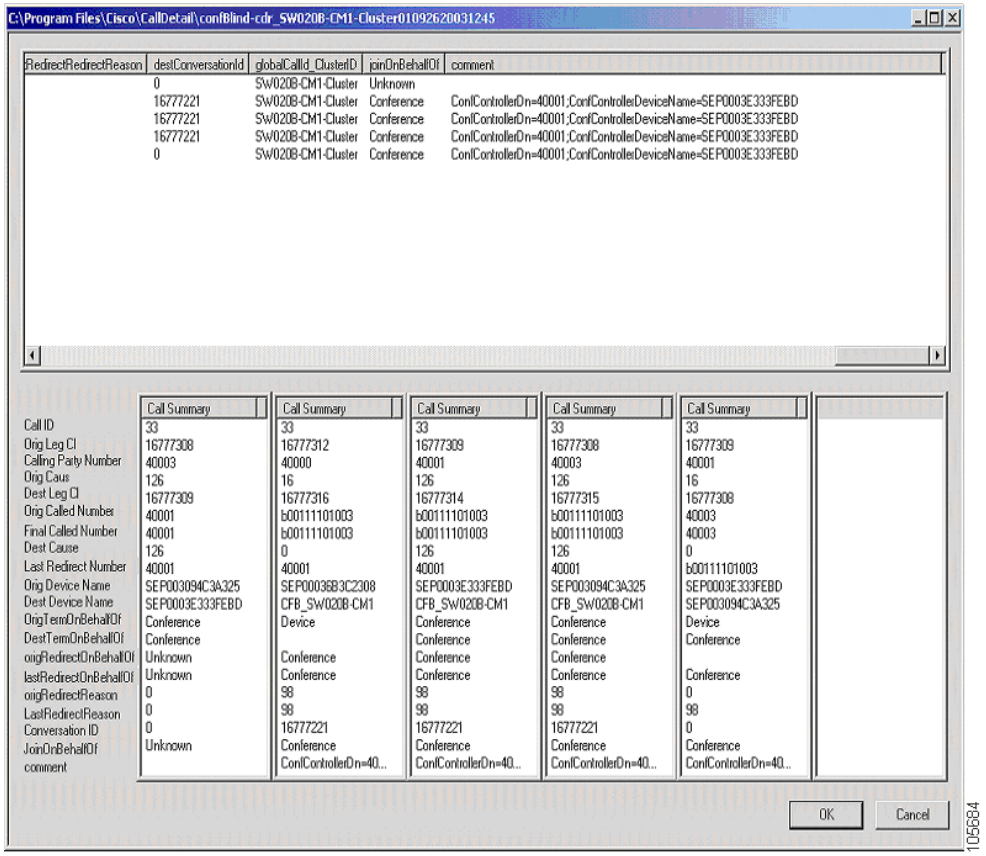
Blind Conference

Figure 6 shows a view of a blind conference call with the following scenario:

- 40003 calls 40001.
- 40001 presses the conference button and calls 40000.
- 40001 presses the conference button to complete the transfer.
- 40003 hangs up first, and 40000 and 40001 are joined in a direct call.

Note The comment field identifies the controller.

Figure 6 Blind Conference Call



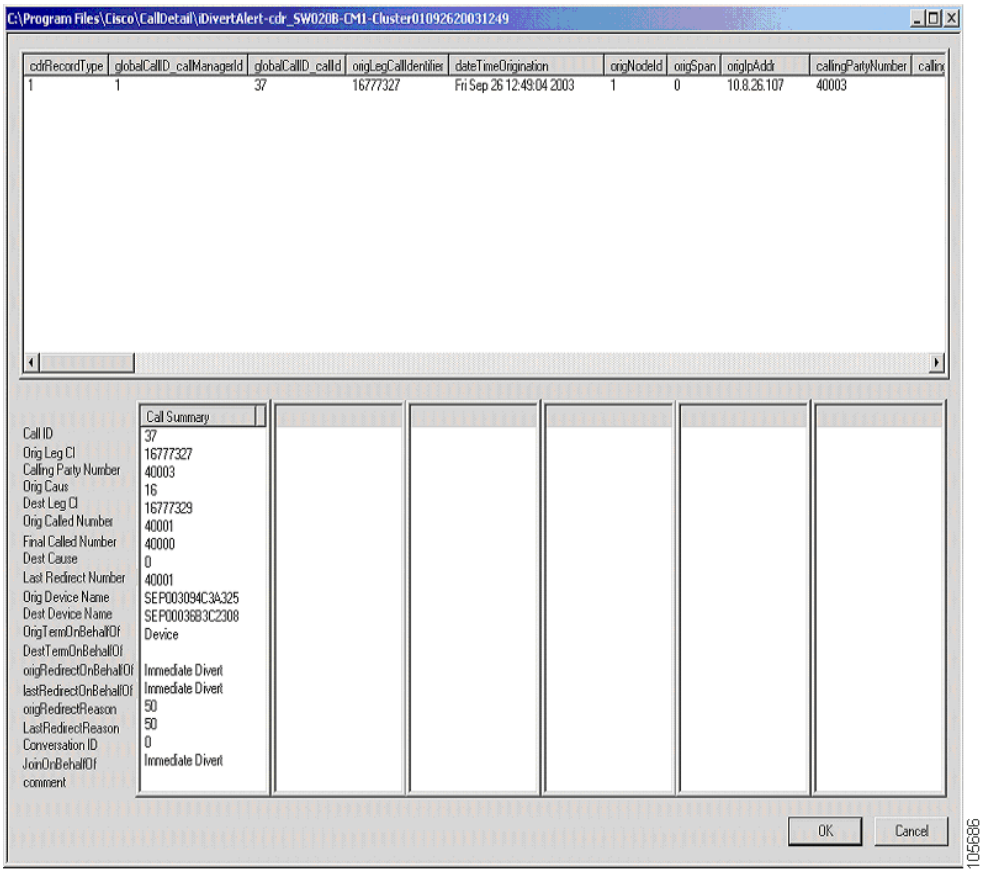
Immediate Divert (IDivert) During Alerting

Figure 7 shows a view of IDivert during Alerting with the following scenario:

- 40003 calls 40001.
- 40001 presses the IDivert button, and the call diverts to 40000.

Note If IDivert redirects the call in the Alerting state, only one CDR gets generated.

Figure 7 IDivert During Alerting



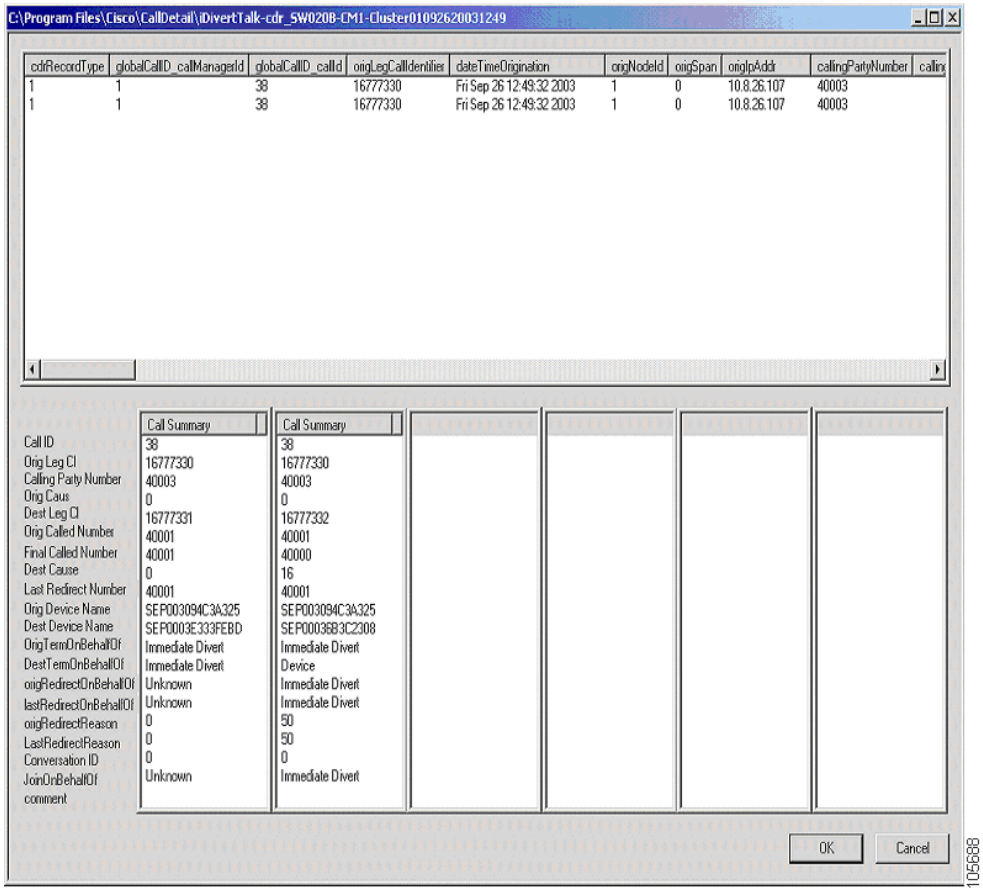
IDivert During Connected

Figure 8 shows a view of IDivert during Connected with the following scenario:

- 40003 calls 40001.
- 40001 answers the call.
- 40001 presses the IDivert button, and the call diverts to 40000.

Note If the call gets connected and redirected by IDivert, two CDRs get generated.

Figure 8 IDivert During Connected



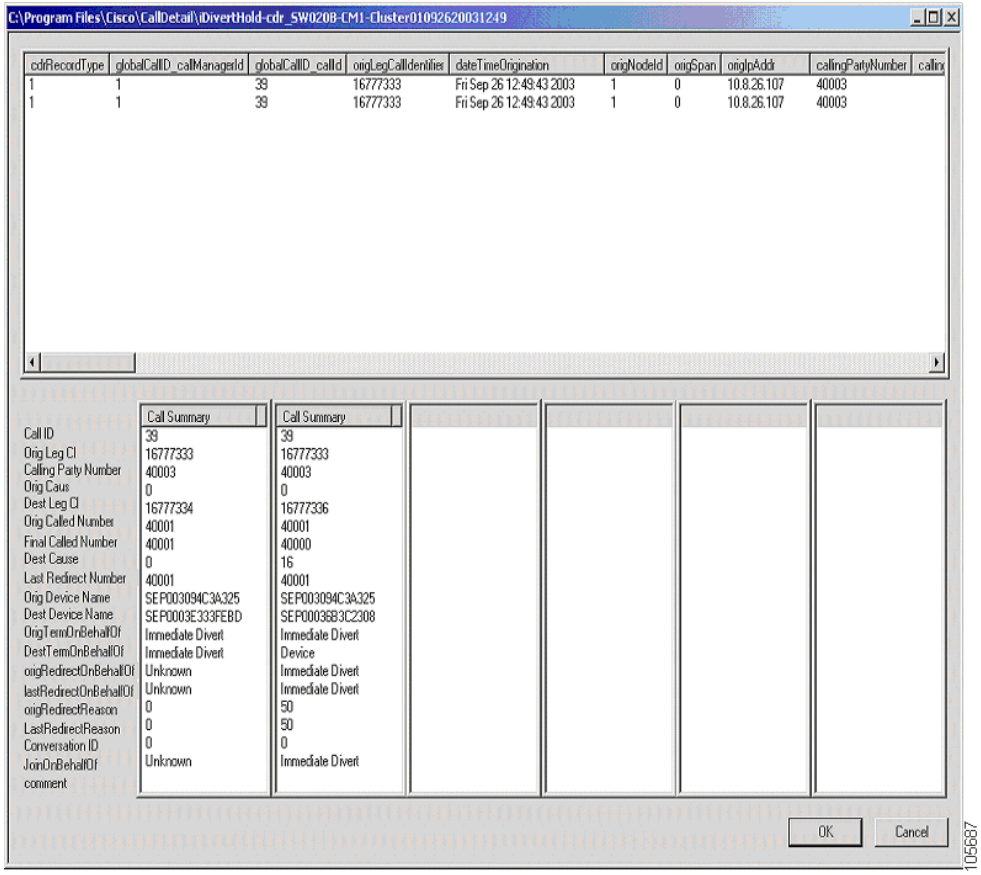
IDivert During Hold

Figure 9 shows a view of IDivert during Hold with the following scenario:

- 40003 calls 40001.
- 40001 answers the call and puts 40003 on hold.
- 40001 presses the IDivert button, and the call diverts to 40000.

Note If the call gets connected and redirected by IDivert, two CDRs get generated.

Figure 9 IDivert During Hold



Barge

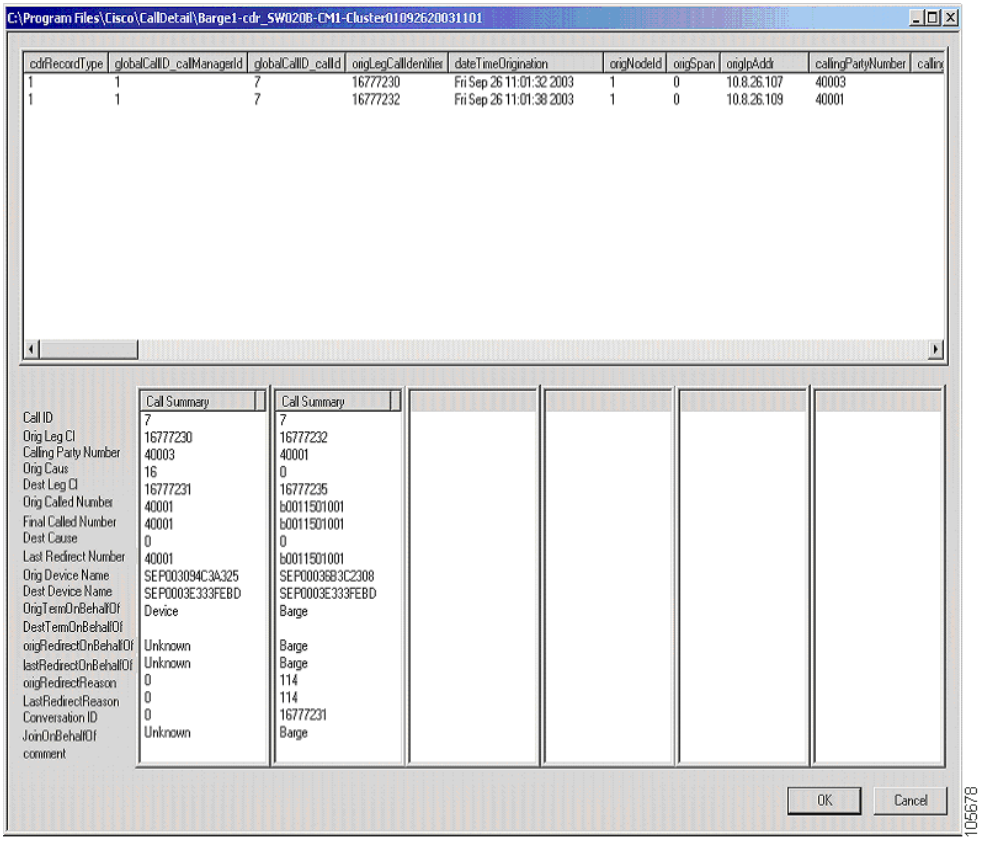
Example 1

Figure 10 shows a view of Barge with the following scenario:

- 40003 calls 40001.
- 40001 answers the call.
- 40001' (shared line) on another phone presses the Barge button.
- 40003 hangs up.

Note The conversationID field links back to the Call Identifier (CI) of the barged call.

Figure 10 Barge Example 1



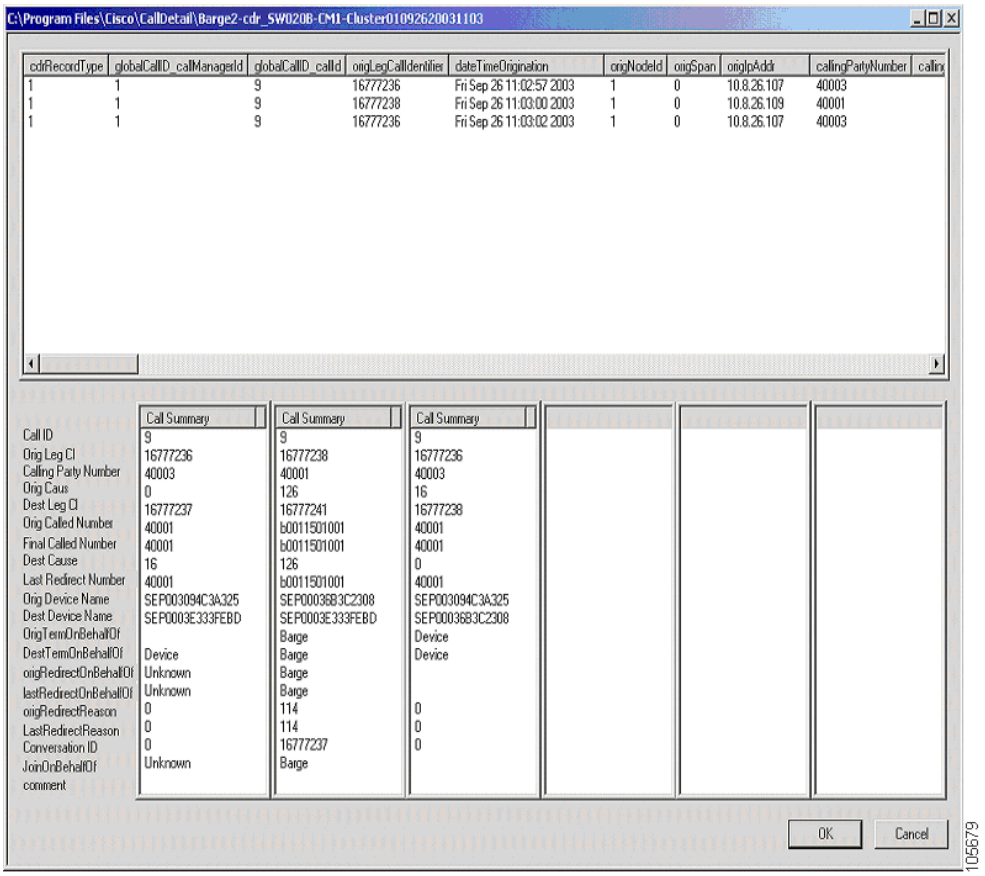
Example 2

Figure 11 shows a view of Barge with the following scenario:

- 40003 calls 40001.
- 40001 answers the call.
- 40001' (shared line) on another phone presses the Barge button.
- 40001 hangs up first.

Note The conversationID field links back to the Call Identifier (CI) of the barged call.

Figure 11 Barge Example 2



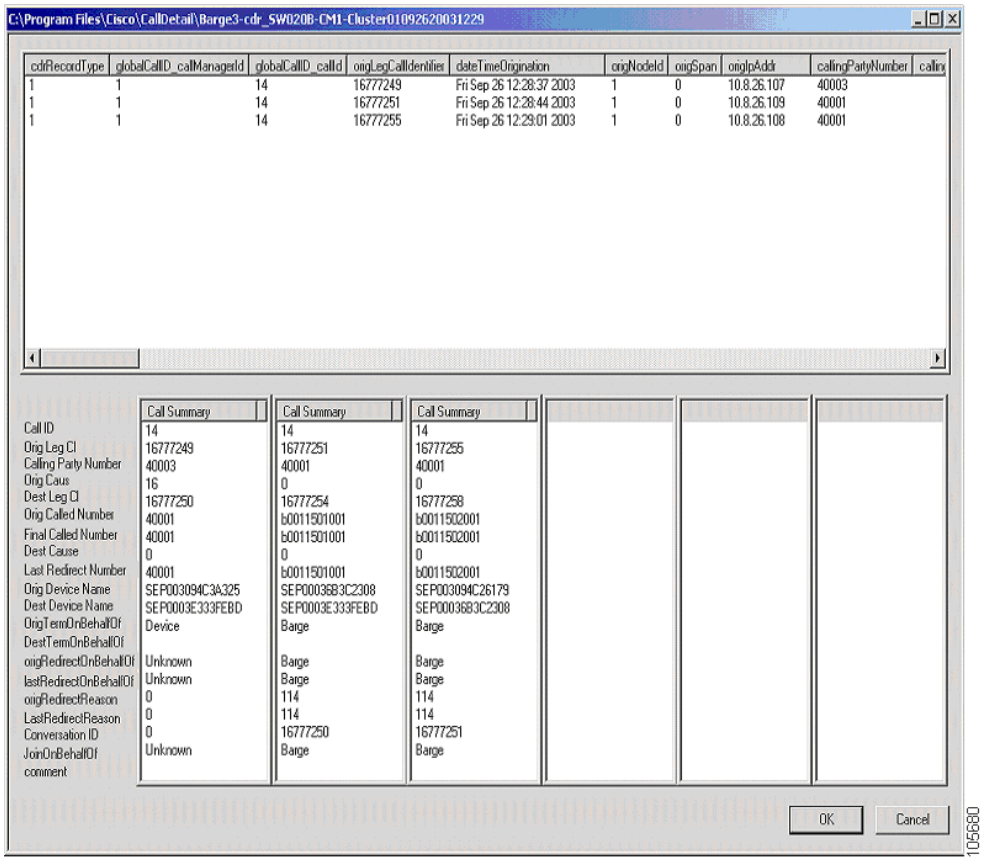
Example 3

Figure 12 shows a view of Barge with the following scenario:

- 40003 calls 40001.
- 40001 answers the call.
- 40001' (shared line) on another phone presses the Barge button.
- 40001'' (another shared line) selects 40001' and presses the Barge button.
- 40003 hangs up first.

Note The conversationID field links back to the Call Identifier (CI) of the barged call.

Figure 12 Barge Example 3



cBarge

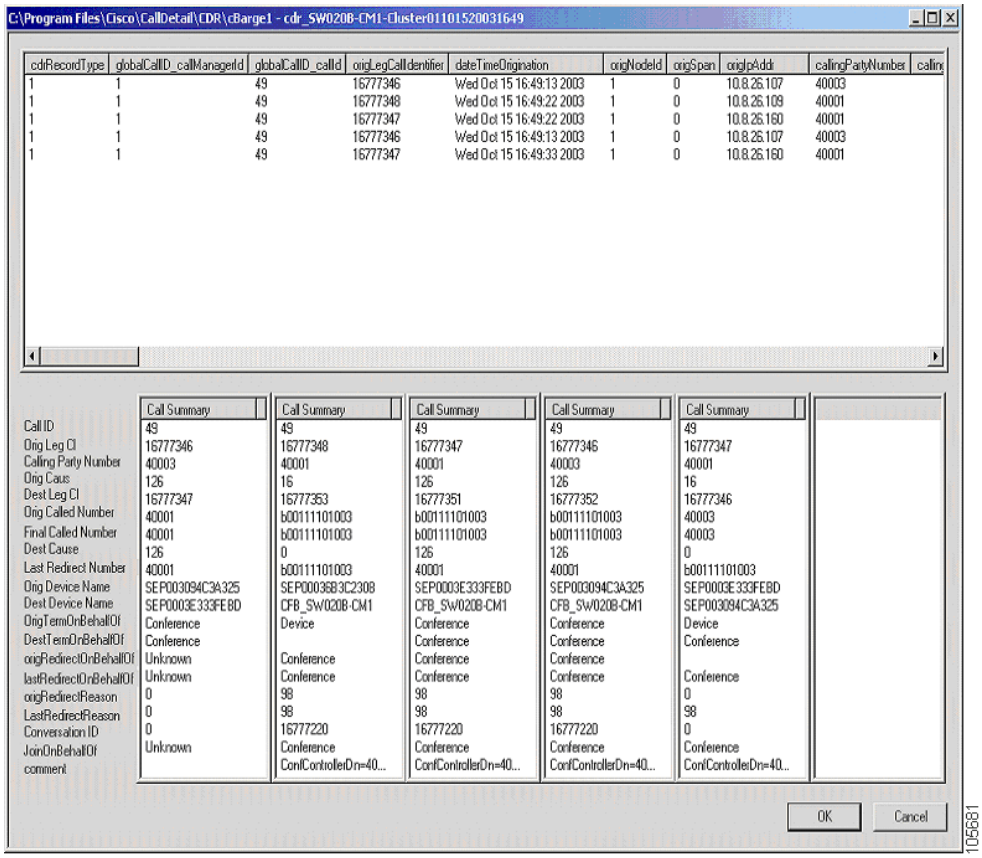
Example 1

Figure 13 shows a view of cBarge with the following scenario:

- 40003 calls 40001.
- 40001 answers the call.
- 40001' (shared line) on another phone presses the cBarge button.

Note The comment field identifies the controller.

Figure 13 cBarge Example 1



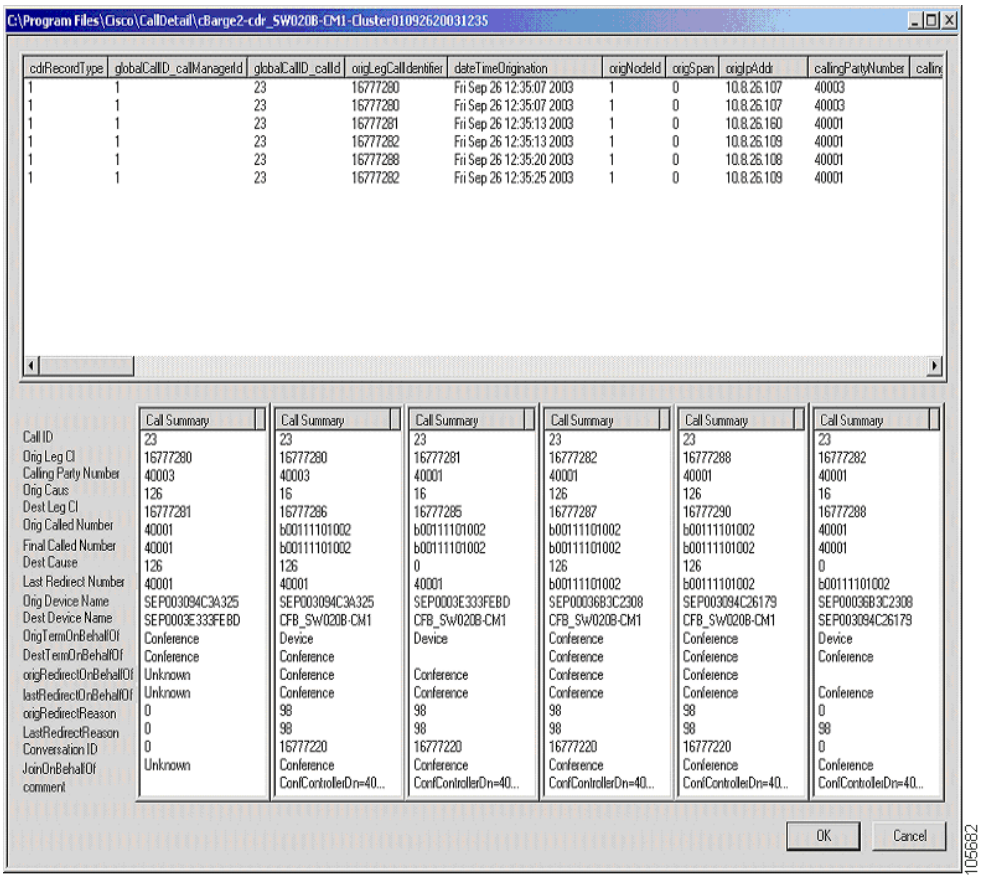
Example 2

Figure 14 shows a view of cBarge with the following scenario:

- 40003 calls 40001.
- 40001 answers the call.
- 40001' (shared line) on another phone presses the cBarge button.
- 40001" (another shared line) on another phone presses the cBarge button.

Note The comment field identifies the controller.

Figure 14 cBarge Example 2



Known Issues

The Cisco CallManager 4.2 has several know issues with CDR data, which are listed in this section.

Ad Hoc Conferences

During an ad hoc conference, all CDRs show call legs into the bridge, regardless of the actual direction of the call. You cannot determine whether a participating call leg is incoming or outgoing by examining the CDRs connected to the conference bridge. In order to determine the direction of the call, the original call and the consultation calls need to be examined.

End-of-Call Records

The Cisco CallManager only generates end-of-call records. You cannot see records of calls in progress.

On-Net vs Off-Net

You may have difficultly determining whether a call stays completely on the IP network or at least internal to the local system. One way you can verify this information is to check the device type of both ends of the call. If both are phones, you can assume that the call stayed on-net. If one device is a gateway, you must consider the following information. If the gateway is an analog access type of device with a POTS or station port, the call may have gone to a local analog phone or to the PSTN. You must look at the number dialed and the dial plan to determine whether the call went off-net.

Off-Net Digits Dialed

If a call is placed out of a gateway, the digits dialed to get to the gateway may not be the digits sent to the PSTN. The gateway may modify the directory number further. The Cisco CallManager does not receive the modified number, and the CDR does not reflect the actual digits that are sent off-net.

Troubleshooting

This section covers an issue that is related to CDRs.

CDRs Fail to Insert

Symptom

A third-party CDR application is installed, and CDRs fail to insert.

Probable Cause

The third-party CDR application may cause this issue.

Corrective Action

Alter all insert triggers for the third-party application and change them to update triggers instead of insert triggers. This will help determine whether the issue is due to the third-party application.

Related Documentation

The following documents contain additional information related to CDRs:

- *Cisco CallManager Serviceability Administration*
- *Cisco CallManager Serviceability System Guide*
- *Cisco CallManager System Guide*

Obtaining Documentation

Cisco documentation and additional literature are available on Cisco.com. Cisco also provides several ways to obtain technical assistance and other technical resources. These sections explain how to obtain technical information from Cisco Systems.

Cisco.com

You can access the most current Cisco documentation at this URL:

http://www.cisco.com/univercd/home/home.htm

You can access the Cisco website at this URL:

http://www.cisco.com

You can access international Cisco websites at this URL:

http://www.cisco.com/public/countries_languages.shtml

Documentation DVD

Cisco documentation and additional literature are available in a Documentation DVD package, which may have shipped with your product. The Documentation DVD is updated regularly and may be more current than printed documentation. The Documentation DVD package is available as a single unit.

Registered Cisco.com users (Cisco direct customers) can order a Cisco Documentation DVD (product number DOC-DOCDVD=) from the Ordering tool or Cisco Marketplace.

Cisco Ordering tool:

http://www.cisco.com/en/US/partner/ordering/

Cisco Marketplace:

http://www.cisco.com/go/marketplace/

Ordering Documentation

You can find instructions for ordering documentation at this URL:

http://www.cisco.com/univercd/cc/td/doc/es_inpck/pdi.htm

You can order Cisco documentation in these ways:

- Registered Cisco.com users (Cisco direct customers) can order Cisco product documentation from the Ordering tool:

  http://www.cisco.com/en/US/partner/ordering/

- Nonregistered Cisco.com users can order documentation through a local account representative by calling Cisco Systems Corporate Headquarters (California, USA) at 408 526-7208 or, elsewhere in North America, by calling 1 800 553-NETS (6387).

Documentation Feedback

You can send comments about technical documentation to bug-doc@cisco.com.

You can submit comments by using the response card (if present) behind the front cover of your document or by writing to the following address:

Cisco Systems
Attn: Customer Document Ordering
170 West Tasman Drive
San Jose, CA 95134-9883

We appreciate your comments.

Cisco Product Security Overview

This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at: http://www.cisco.com/wwl/export/crypto/tool/stqrg.html.

If you require further assistance please contact us by sending email to export@cisco.com.

Cisco provides a free online Security Vulnerability Policy portal at this URL:

http://www.cisco.com/en/US/products/products_security_vulnerability_policy.html

From this site, you can perform these tasks:

- Report security vulnerabilities in Cisco products.
- Obtain assistance with security incidents that involve Cisco products.
- Register to receive security information from Cisco.

A current list of security advisories and notices for Cisco products is available at this URL:

http://www.cisco.com/go/psirt

If you prefer to see advisories and notices as they are updated in real time, you can access a Product Security Incident Response Team Really Simple Syndication (PSIRT RSS) feed from this URL:

http://www.cisco.com/en/US/products/products_psirt_rss_feed.html

Reporting Security Problems in Cisco Products

Cisco is committed to delivering secure products. We test our products internally before we release them, and we strive to correct all vulnerabilities quickly. If you think that you might have identified a vulnerability in a Cisco product, contact PSIRT:

- Emergencies—security-alert@cisco.com
- Nonemergencies—psirt@cisco.com

Tip We encourage you to use Pretty Good Privacy (PGP) or a compatible product to encrypt any sensitive information that you send to Cisco. PSIRT can work from encrypted information that is compatible with PGP versions 2.*x* through 8.*x*.

Never use a revoked or an expired encryption key. The correct public key to use in your correspondence with PSIRT is the one that has the most recent creation date in this public key server list:

http://pgp.mit.edu:11371/pks/lookup?search=psirt%40cisco.com&op=index&exact=on

In an emergency, you can also reach PSIRT by telephone:

- 1 877 228-7302
- 1 408 525-6532

Obtaining Technical Assistance

For all customers, partners, resellers, and distributors who hold valid Cisco service contracts, Cisco Technical Support provides 24-hour-a-day, award-winning technical assistance. The Cisco Technical Support Website on Cisco.com features extensive online support resources. In addition, Cisco Technical Assistance Center (TAC) engineers provide telephone support. If you do not hold a valid Cisco service contract, contact your reseller.

Cisco Technical Support Website

The Cisco Technical Support Website provides online documents and tools for troubleshooting and resolving technical issues with Cisco products and technologies. The website is available 24 hours a day, 365 days a year, at this URL:

http://www.cisco.com/techsupport

Access to all tools on the Cisco Technical Support Website requires a Cisco.com user ID and password. If you have a valid service contract but do not have a user ID or password, you can register at this URL:

http://tools.cisco.com/RPF/register/register.do

Note Use the Cisco Product Identification (CPI) tool to locate your product serial number before submitting a web or phone request for service. You can access the CPI tool from the Cisco Technical Support Website by clicking the **Tools & Resources** link under Documentation & Tools. Choose **Cisco Product Identification Tool** from the Alphabetical Index drop-down list, or click the **Cisco Product Identification Tool** link under Alerts & RMAs. The CPI tool offers three search options: by product ID

or model name; by tree view; or for certain products, by copying and pasting **show** command output. Search results show an illustration of your product with the serial number label location highlighted. Locate the serial number label on your product and record the information before placing a service call.

Submitting a Service Request

Using the online TAC Service Request Tool is the fastest way to open S3 and S4 service requests. (S3 and S4 service requests are those in which your network is minimally impaired or for which you require product information.) After you describe your situation, the TAC Service Request Tool provides recommended solutions. If your issue is not resolved using the recommended resources, your service request is assigned to a Cisco TAC engineer. The TAC Service Request Tool is located at this URL:

http://www.cisco.com/techsupport/servicerequest

For S1 or S2 service requests or if you do not have Internet access, contact the Cisco TAC by telephone. (S1 or S2 service requests are those in which your production network is down or severely degraded.) Cisco TAC engineers are assigned immediately to S1 and S2 service requests to help keep your business operations running smoothly.

To open a service request by telephone, use one of the following numbers:

Asia-Pacific: +61 2 8446 7411 (Australia: 1 800 805 227)
EMEA: +32 2 704 55 55
USA: 1 800 553-2447

For a complete list of Cisco TAC contacts, go to this URL:

http://www.cisco.com/techsupport/contacts

Definitions of Service Request Severity

To ensure that all service requests are reported in a standard format, Cisco has established severity definitions.

Severity 1 (S1)—Your network is “down,” or there is a critical impact to your business operations. You and Cisco will commit all necessary resources around the clock to resolve the situation.

Severity 2 (S2)—Operation of an existing network is severely degraded, or significant aspects of your business operation are negatively affected by inadequate performance of Cisco products. You and Cisco will commit full-time resources during normal business hours to resolve the situation.

Severity 3 (S3)—Operational performance of your network is impaired, but most business operations remain functional. You and Cisco will commit resources during normal business hours to restore service to satisfactory levels.

Severity 4 (S4)—You require information or assistance with Cisco product capabilities, installation, or configuration. There is little or no effect on your business operations.

Obtaining Additional Publications and Information

Information about Cisco products, technologies, and network solutions is available from various online and printed sources.

- Cisco Marketplace provides a variety of Cisco books, reference guides, and logo merchandise. Visit Cisco Marketplace, the company store, at this URL:

  http://www.cisco.com/go/marketplace/

- *Cisco Press* publishes a wide range of general networking, training and certification titles. Both new and experienced users will benefit from these publications. For current Cisco Press titles and other information, go to Cisco Press at this URL:

  http://www.ciscopress.com

- *Packet* magazine is the Cisco Systems technical user magazine for maximizing Internet and networking investments. Each quarter, Packet delivers coverage of the latest industry trends, technology breakthroughs, and Cisco products and solutions, as well as network deployment and troubleshooting tips, configuration examples, customer case studies, certification and training information, and links to scores of in-depth online resources. You can access Packet magazine at this URL:

  http://www.cisco.com/packet

- *iQ Magazine* is the quarterly publication from Cisco Systems designed to help growing companies learn how they can use technology to increase revenue, streamline their business, and expand services. The publication identifies the challenges facing these companies and the technologies to help solve them, using real-world case studies and business strategies to help readers make sound technology investment decisions. You can access iQ Magazine at this URL:

  http://www.cisco.com/go/iqmagazine

- *Internet Protocol Journal* is a quarterly journal published by Cisco Systems for engineering professionals involved in designing, developing, and operating public and private internets and intranets. You can access the Internet Protocol Journal at this URL:

  http://www.cisco.com/ipj

- World-class networking training is available from Cisco. You can view current offerings at this URL:

  http://www.cisco.com/en/US/learning/index.html

This document is to be used in conjunction with the documents listed in the "Related Documentation" section.