



Call Detail and Call Management Records

This appendix provides detailed information about Call Detail Records (CDR) and Call Management Records (CMR).

CDR records are written to a database for use in post-processing activities. These activities include many functions but this appendix will focus primarily on billing and network analysis.

The database is a Microsoft SQL Server 7.0 database. Access to the database can be made via Open DataBase Connectivity (ODBC).

Access is provided to all tables in the database in a read-only fashion, and to the CDR and CMR tables in a read or write fashion.

To use CDR record data, you may want to read other tables in the database in an effort to obtain information about the type of device the CDR describes. This correlation between devices in the device table and the IP address listed in the CDR record is not straightforward and is listed as a known issue later in this appendix.

This appendix contains the following topics:

- [Writing Records, page D-2](#)
- [Reading Records, page D-2](#)
- [Removing Records, page D-3](#)
- [Table Schema, page D-4](#)
- [Issues, page D-4](#)
- [Field Data Conversions, page D-4](#)

- [Call Records Logged By Call Type](#), page D-20
- [Call Management Records Logged By Call Type](#), page D-22
- [Codec Types \(Compression / Payload types\)](#), page D-23
- [Cause Codes](#), page D-24
- [Alarms](#), page D-28

Writing Records

Cisco CallManager writes CDR records to the SQL database as calls are made in a manner consistent with the configuration of each individual Cisco CallManager. This configuration is made using the Service Parameters screen in Cisco CallManager Administration.

All records are written to the primary database for a cluster. If the primary database is not available, then they will be written to any of the other backup databases. Once the primary database becomes available, then writing new records will continue on the primary database and the locally written records will be moved to the primary.

Reading Records

The easiest way to read data from the SQL database may be to use ODBC. A good connection string would look like:

```
DRIVER={SQL Server};SERVER=machineX;DATABASE=CCM0300
```

**Note**

Be sure to use the correct database name. If you have installed a Cisco CallManager Release 3.1(1) version over an existing installation, then the database might be migrated if called for by the new installation. In this case, the old database will still exist, and the new database will also exist. The names will differ by adding one to the number of the name. For instance, the original name is CCM0300. After a migration, the newer database name will be CCM0311. You should use the highest number database.

Refer to the Cisco CallManager Release 3.1 documentation available online at the following location:

http://www.cisco.com/univercd/cc/td/doc/product/voice/c_callmg/3_1/index.htm

The primary database (machine and name) currently in use by the cluster can be found by clicking on Details in Cisco CallManager Administration (click Help to reach the Welcome screen where Details is located). The registry on machines hosting a database can also be checked. Look at the registry key: \\HKEY_LOCAL_MACHINE\Software\Cisco Systems Inc.\DBL for the item called DBConnection0. This string item contains a connection string similar to that shown above with the machine name and database name of the primary database.

Access is controlled by use of SQL Users. The following table specifies the UserID and password that should be used when accessing the Cisco CallManager database.

Tables	SQL UserID	Password	Capability
CallDetailRecord	CiscoCCMCDR	dipsy	Read/Write
CallDetailRecordDiagnostic			
(Other)	CiscoCCMReader	cowboys	Read Only

Removing Records

Because Cisco CallManager relies on third-party applications to post-process the CDR data, you should remove the CDR data when all applications are through with the data.

If CDR records accumulate to a configured maximum (10,000,000 CDR records), then the oldest CDR records will be removed along with related CMR records once a day.

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

Table Schema

Detailed information about the format and use of each field in the CDR is provided in [Table D-1](#).

The main tables you should use are the CallDetailRecord table, which holds CDR records, and the CallDetailRecordDiagnostic table, which holds CMR records. The CallDetailRecord table is related to the CallDetailRecordDiagnostic table via the two GlobalCallID columns, GlobalCallID_callManagerId and GlobalCallID_callId. There may be more than one CMR per CDR.

The CallDetailRecord table holds information about the endpoints of the call and other call control/routing aspects of the call. The CallDetailRecordDiagnostic table holds information about the quality of the streamed audio of the call.

Issues

Cisco CallManager Release 3.1(1) may have issues with the CDR data. Contact Cisco TAC or refer to the Release Notes for Cisco CallManager 3.1(1) at the following location:

http://www.cisco.com/univercd/cc/td/doc/product/voice/c_callmg/3_1/rel_note/index.htm

Field Data Conversions

There are some fields that require conversion from decimal format to another format for displays. This appendix defines their values, and how to convert them or where to get information on how to convert them.

Time Values

All time values are represented as unsigned 32 bit integers. This unsigned integer value is displayed from the database as a signed integer.

This field is a `time_t` value that is obtained from the Windows NT (2000) system routines. The value is a coordinated universal time (UTC) value, and represents the number of seconds since Midnight (00:00:00) Jan. 1, 1970.

Deciphering the Time Stamp

Using Microsoft Excel, you can write a formula to make converting this time stamp a little easier. If the value is in cell A1, you can make another cell:

```
=A1/86400+DATE(1970,1,1)
```

There are 86400 seconds in a day. Then format the resulting cell as a date/time field in Excel.

IP Addresses

All IP addresses are stored in the system 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 hexadecimal number (taking into consideration that it is really an unsigned number). The 32bit hexadecimal value represents four bytes. The four bytes are in reverse order (Intel standard). To get the IP address, reverse the order of the bytes, and convert each byte to a decimal number. The resulting four bytes represent the 4-byte fields of the IP address in dotted notation.

**Note**

The database will display a negative number when the low byte of the IP address has the most significant bit set.

Converting IP Addresses

For example: IP Address 192.168.18.188 would be displayed as follows

Database Display = -1139627840.

This converts to a hexadecimal value of 0xBC12A8C0.

Reverse the hexadecimal bytes = C0A812BC

C0 A8 12 BC

Bytes Converted from hexadecimal to Decimal = 192 168 18 188 which would be displayed as 192.168.18.188.

Example2: IP Address 192.168.18.59

Database Display = 991078592

This converts to a hexadecimal value of 0x3B12A8C0

Reverse Byte order = C0A8123B

C0 A8 12 3B

Bytes Converted from hexadecimal to Decimal = 192 168 18 59 which would be displayed as 192.168.18.59.

CDR Field Definition

[Table D-1](#) provides field definitions for CDRs.

Table D-1 CDR Field Definitions

Field	Definition
cdrRecordType	Type of this record unsigned integer Specifies the type of this specific record. It could be a Start call record(0), End call record(1), or a CMR record(2).
globalCallIdentifier	Global Call Identifier The Global Call Identifier consists of two fields that are both unsigned integers. The values must be treated as unsigned integers. The two fields are: Unsigned integer GlobalCallID_CallID This is the call identifier that is assigned to the entire call. All records associated with a standard call will have the same global call identifier.
origLegCallIdentifier	Origination leg call identifier unsigned integer This is a unique identifier that is used to track the origination leg of a call. It is unique within a cluster.

Table D-1 CDR Field Definitions (continued)

Field	Definition
dateTimeOrigination	Date/time of call origination unsigned integer This represents the time that the device originating the call went off hook, or the time that an outside call was first recognized by the system (it received the Setup message). The value is a coordinated universal time (UTC) value, and represents the number of seconds since Midnight (00:00:00) Jan. 1, 1970.
origNodeId	Originator's node ID unsigned integer This field represents the node within the Cisco CallManager cluster where the call originator was registered at the time of this call.
origSpan	Originator's span or port unsigned integer This field contains the originator's port or span number if the call originated through a gateway. If not, this field contains zero (0).
callingPartyNumber	Calling party number up to 25 characters This is the directory number of the device from which the call originated.
origIpPort	Calling party's IP port unsigned integer This field contains the IP Port of the device from which the call originated.

Table D-1 CDR Field Definitions (continued)

Field	Definition
origIpAddr	Calling party's IP address unsigned integer This field contains the IP address of the device from which the call originated.
originalCallingPartyNumberPartition	Calling party's partition up to 50 characters This field contains the Partition associated with the calling party.
origCause_Location	SDN location value unsigned integer This field contains the location value from the Cause information Element.
origCause_Value	Calling party cause Of call termination unsigned integer This cause represents why the call to the originating device was terminated. In the case of transfers, forwards, and so on, the cause of call termination may be different for the originating device and the termination device. Thus, there are two cause fields associated with each call. Usually they will be the same.
origMediaTransportAddress_IP	The IP address for the originator's media connection unsigned integer This is the destination IP Address to which the Media Stream from the originator was connected.

Table D-1 CDR Field Definitions (continued)

Field	Definition
origMediaTransportAddress_Port	<p>The port for the originator's media connection</p> <p>unsigned integer</p> <p>This is the destination port to which the Media Stream from the originator was connected.</p>
origMediaCap_payloadCapability	<p>The codec type used by the originator</p> <p>unsigned integer</p> <p>This field contains the Codec type (compression or payload type) that the originator used on the sending side during this call. It may be different than the codec type used on its receiving side.</p>
origMediaCap_maxFramesPerPacket	<p>The number of milliseconds of data per packet</p> <p>unsigned integer</p> <p>This field contains the number of milliseconds of data per packet sent to the destination, by the originator of this call. The actual data size depends on the codec type being used to generate the data.</p>
origMediaCap_g723BitRate	<p>The bit rate to be used by G.723</p> <p>unsigned integer</p> <p>Defines the bit rate to be used by G.723. There are two bit rate values. They are: 1 = 5.3K bit rate, and 2 = 6.3K bit rate.</p>

Table D-1 CDR Field Definitions (continued)

Field	Definition
lastRedirectDn	<p>Directory number of the party that last redirected this call</p> <p>up to 25 characters</p> <p>This is the directory number of the last device that redirected this call. This field applies only to calls that were redirected, such as conference calls, call forwarded calls, and so on.</p>
lastRedirectDnPartition	<p>Partition of the phone that last redirected this call</p> <p>up to 50 characters</p> <p>This is the Partition of the last device that redirected this call. This field applies only to calls that were redirected such as conference calls, call forwarded calls, and so on.</p>
destLegIdentifier	<p>The call identifier for the destination leg of the call</p> <p>unsigned integer</p> <p>This is a unique identifier that is used to track the destination leg of this call. It is unique within a cluster.</p>
destNodeId	<p>The node identifier for the node where the destination of the call was registered</p> <p>unsigned integer</p> <p>The node within the Cisco CallManager cluster where the destination device was registered at the time of this call.</p>

Table D-1 CDR Field Definitions (continued)

Field	Definition
dest Span	<p>The destination span or port unsigned integer</p> <p>This field contains the destination port or span number if the call was terminated through a gateway. If not, this field contains a (0) zero.</p>
destIpAddr	<p>The IP address to which the call was delivered unsigned integer</p> <p>This field contains the IP address of the signaling connection on the device that terminated the call.</p>
destIpPort	<p>The IP port to which the call was delivered unsigned integer</p> <p>This field contains the IP port of the signaling connection on the device that terminated the call.</p>
originalCalledPartyNumber	<p>The destination received from the call originator up to 25 characters</p> <p>This field contains the Directory Number to which the call was originally extended based on the digits dialed by the originator of the call. If the call completes normally (meaning it was not forwarded), this Directory Number should always be the same as the finalCalledPartyNumber. If the call was forwarded, this field contains the original destination of the call before it was forwarded.</p>

Table D-1 CDR Field Definitions (continued)

Field	Definition
originalCalledPartyNumberPartition	<p>Called party's partition up to 50 characters</p> <p>This field contains the partition associated with the called party.</p>
finalCalledPartyNumber	<p>The destination to which the call was delivered up to 25 characters</p> <p>This field contains the Directory Number to which the call was actually extended. If the call completes normally (meaning it was not forwarded), this Directory Number should always be the same as the originalCalledPartyNumber. If the call was forwarded, this field contains the Directory Number of the final destination of the call after all forwards were completed.</p>
finalCalledPartyNumberPartition	<p>The partition associated with the final destination of the call. up to 50 characters</p> <p>This field contains the partition associated with the destination to which the call was actually extended. In a normal call, this field should be the same as originalCalledPartyNumberPartition. If the call was forwarded, this field contains the partition of the final destination of the call after all forwards were completed.</p>

Table D-1 CDR Field Definitions (continued)

Field	Definition
destCause_location	Called party cause location unsigned integer This is the ISDN Location value from the Cause Information Element.
destCause_value	Called party cause of call termination unsigned integer This cause describes why the device was terminated. In the case of transfers, forwards, and so on, the cause of call termination may be different for the recipient of the call and the originator of the call. Thus, there are two cause fields associated with each call. Usually they will be the same. When an attempt is made to extend a call to a busy device that is forwarded, the cause code will reflect Busy even though the call was connected to a forward destination.
destMediaTransportAddress_IP	The IP address for the destination outgoing media connection unsigned integer This is the origination IP Address from which the Media Stream from the destination was connected.
origMediaTransportAddress_Port	The port for the destination outgoing media connection unsigned integer This is the originator's port from which the Media Stream from the destination was connected.

Table D-1 CDR Field Definitions (continued)

Field	Definition
destMediaCap_payloadCapability	<p>The codec type used by the destination on the sending side.</p> <p>unsigned integer</p> <p>This field contains the Codec type (compression or payload type) that the destination used on its sending side during this call. It may be different than the codec type used on its receiving side.</p>
destMediaCap_maxFramesPerPacket	<p>The number of milliseconds of data per packet</p> <p>unsigned integer</p> <p>This field contains the number of milliseconds of data per packet sent to the originator, by the destination of this call. The actual data size depends on the codec type being used to generate the data.</p>
destMediaCap_g723BitRate	<p>The bit rate to be used by G.723</p> <p>unsigned integer</p> <p>Defines the bit rate to be used by G.723. There are two bit rate values. They are: 1 = 5.3K bit rate, and 2 = 6.3K bit rate.</p>

Table D-1 CDR Field Definitions (continued)

Field	Definition
dateTimeConnect	<p>Date/time of connect</p> <p>unsigned integer</p> <p>This is the date and time that the call was connected between the originating and terminating devices. This is the date and time that the call was connected between the originating and terminating devices. The value is a coordinated universal time (UTC) value, and represents the number of seconds since Midnight (00:00:00) Jan. 1, 1970.</p>
dateTimeDisconnect	<p>Date/time of disconnect</p> <p>unsigned integer</p> <p>This is the time that the call was disconnected between the originating and terminating devices, or when the call was torn down even if it was never connected. The value is a coordinated universal time (UTC) value, and represents the number of seconds since Midnight (00:00:00) Jan. 1, 1970.</p>
duration	<p>Call duration</p> <p>This is the number of seconds that the call was connected. It is the difference between the date/time of connect and the date/time of disconnect.</p>

Table D-2 provides field definitions for CMRs (diagnostic CDRs).

Table D-2 CMR Field Definitions

Field	Definition
cdrRecordType	Type of this record unsigned integer Specifies the type of this specific record. It will be set to CMR record.
globalCallIdentifier	Global Call Identifier for this call The Global Call Identifier consists of two fields which are both unsigned integers. The values must be treated as unsigned integers. The two fields are: Unsigned integer GlobalCallID_CallID Unsigned integer GlobalCallID_CallManagerID This is the call identifier that is assigned to the entire call. All records associated with a standard call will have the same global call identifier.
nodeID	The Cisco CallManager node identifier The node within the Cisco CallManager cluster where this record was generated.
callIdentifier	Call Identifier unsigned integer This is a call leg identifier that identifies to which call leg this record pertains.
directoryNum	Directory number used on this call This is the directory number of the device from which these diagnostics were collected.

Table D-2 CMR Field Definitions (continued)

Field	Definition
directoryNumPartition	The partition associated with the directory number This is the partition of the directory number in this record.
dateTimeStamp	Date/time of call termination This represents the approximate time that the device went on hook. The time is put into the record when the phone responds to a request for diagnostic information. This is a time_t value.
numberPacketsSent	Number of packets sent The total number of RTP data packets transmitted by the device since starting transmission on this connection. The value is zero if the connection was set in receive only mode.
numberOctetsSent	Number of octets (bytes) of data sent to the other party The total number of payload octets (that is, not including header or padding) transmitted in RTP data packets by the device since starting transmission on this connection. The value is zero if the connection was set in receive only mode.
numberPacketsReceived	The number of data packets received during this call The total number of RTP data packets received by the device since starting reception on this connection. The count includes packets received from different sources if this is a multicast call. The value is zero if the connection was set in send only mode.

Table D-2 CMR Field Definitions (continued)

Field	Definition
numberOctetsReceived	<p>The number of octets (bytes) of data received during this call</p> <p>The total number of payload octets (that is, not including header or padding) received in RTP data packets by the device since starting reception on this connection. The count includes packets received from different sources, if this is a multicast call. The value is zero if the connection was set in send only mode.</p>
numberPacketsLost	<p>Lost RTP packets during this connection</p> <p>The total number of RTP data packets that have been lost since the beginning of reception. This number is defined as the number of packets expected less the number of packets actually received, where the number of packets received includes any that are late or duplicates. Thus, packets that arrive late are not counted as lost, and the loss may be negative if there are duplicates. The number of packets expected is defined to be the extended last sequence number received, as defined next, less the initial sequence number received. The value is zero if the connection was set in send only mode. (For details, refer to RFC 1889)</p>

Table D-2 CMR Field Definitions (continued)

Field	Definition
jitter	The interarrival jitter during this connection 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 is defined to be 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. Detailed computation algorithms are found in RFC 1889. The value is zero if the connection was set in send only mode.
latency	The latency experienced during this connection The value is an estimate of the network latency, expressed in milliseconds. This is the average value of the difference between the NTP timestamp indicated by the senders of the RTCP messages and the NTP timestamp of the receivers, measured when these messages are received. The average is obtained by summing all the estimates, then dividing by the number of RTCP messages that have been received. (For details refer to RFC 1889)

Call Records Logged By Call Type

Each normal call between two parties logs one CDR End Call record. Each End Call record contains all fields identified above, 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 End Call records may be written.

In addition to the CDR End Call record, there may be up to one CMR record per endpoint involved in a call. In a normal call between two parties each using a Cisco IP Phone, there will be two CMR records written: one for the originator, and one for the destination of the call.

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

Normal Calls (Cisco IP Phone-to-Cisco IP Phone)

Normal calls log three records per call. They are: EndCall plus two diagnostic records, one for each endpoint. In the EndCall record, all fields may contain valid information. The duration will always be non-zero unless the CdrLogCallsWithZeroDurationFlag flag is enabled (set to true). The originalCalledPartyNumber field will contain the same directory number as the finalCalledPartyNumber field.

Abandoned Calls

The logging of calls with zero duration is optional. Normally these records will not be logged. If logging calls with zero duration is enabled, the following things should be noted.

- If the call was abandoned (such as when a phone is taken off hook, and placed back on hook), various fields will not contain data. In this case, the originalCalledPartyNumber, finalCalledPartyNumber, the partitions associated with them, destIpAddr, and the dateTimeConnect fields will be blank. All calls that were not connected will have a duration of zero seconds. When a call is abandoned, the cause code is 0.
- If the user dialed a directory number and then abandoned the call before it was 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 will be blank, and the duration will be zero.

Forwarded or Redirected Calls

The call records for forwarded calls will be the same as those for normal calls except for the originalCalledPartyNumber field, and the originalCalledPartyNumberPartition fields. 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 was forwarded, the finalCalledPartyNumber and finalCalledPartyNumberPartition fields will be different, and will contain the

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

Calls With Busy or Bad Destinations

These calls will be logged as a normal call with all relevant fields containing data. The Called Party Cause field will contain a cause code indicating why the call was not connected, and the Called Party IP and Date/Time Connect fields will be blank. If the originator abandoned the call, the cause will be NO_ERROR (0). The Duration will always be zero seconds. These calls will not be logged unless CdrLogCallsWithZeroDurationFlag is enabled.

Call Management Records Logged By Call Type

Each normal call between two Cisco IP Phones logs exactly two CMR records. Each call CMR record contains all fields identified above. When supplementary services are involved in a call, more than one record may be written. This section describes when diagnostic records are written for different call types in the system.

Normal Calls

Normal calls log exactly two CMR records per call, one for each phone involved in the call. Currently only Cisco IP Phones and MGCP gateways are capable of responding to the diagnostic information request. All fields will contain valid information.

Abandoned Calls

If the call was abandoned (such as when a phone is taken off-hook and placed back on hook), all fields related to streaming data will be blank (zero). This is because no streaming connection was established, and therefore no data was transferred. All records with blank fields will not be logged if the `CdrLogCallsWithZeroDurationFlag` is disabled.

Forwarded Calls

The call records for forwarded calls will be the same as those for normal calls.

Calls With Busy or Bad Destinations

In the normal case, only records that represent calls that were actually connected will be logged. In order to log calls with bad destinations, you must enable `CdrLogCallsWithZeroDurationFlag`. If it is enabled, then all calls will be logged including the case where the user goes off-hook and then on-hook again.

If the calls are logged, they will be logged as normal calls with all relevant fields containing data. There will only be one record per call since the calls were never connected to a destination. The record will be for the originator of the call.

Codec Types (Compression / Payload types)

[Table D-3](#) provides values and descriptions for codec types.

Table D-3 *Codec Description*

Codec	Description
1	NonStandard
2	G711A-law 64k
3	G711A-law 56k
4	G711μ-law 64k

Table D-3 Codec Description (continued)

Codec	Description
5	G711 μ -law 56k
6	G722 64k
7	G722 56k
8	G722 48k
9	G7231
10	G728
11	G729
12	G729AnnexA
13	Is11172AudioCap
14	Is1381AudioCap
15	G729AnnexB
32	Data 64k
33	Data 56k
80	GSM
81	ActiveVoice
82	G726_32K
83	G726_24K
84	G726_16K

Cause Codes

Table D-4 provides a list of cause codes that may appear in the Cause fields.

Table D-4 Cause Code Descriptions

Cause Code	Description
0	No error
1	unallocated (unassigned) number

Table D-4 Cause Code Descriptions (continued)

Cause Code	Description
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	Non-selected user clearing
27	Destination out of order
28	Invalid number format (address incomplete)
29	Facility rejected
30	Response to STATUS ENQUIRY
31	Normal, unspecified
34	No circuit/channel available
38	Network out of order
39	Permanent frame mode connection out of service

Table D-4 Cause Code Descriptions (continued)

Cause 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 CUG (Closed User Group)
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 is available (national use)

Table D-4 Cause Code Descriptions (continued)

Cause Code	Description
79	Service or option not implemented, unspecified
81	Invalid call reference value
82	Identified channel does not exist
83	A suspended call exists, but this call identity does not
84	Call identity in use
85	No call suspended
86	Call having the requested call identity has been cleared
87	user not member of CUG (Closed User Group)
88	Incompatible destination
90	Destination number missing and DC not subscribed
91	Invalid transit network selection (national use)
95	Invalid message, unspecified
96	Mandatory information element is missing
97	Message type non-existent or not implemented
98	Message is not compatible with the call state, or the message type is non-existent or not implemented
99	An information element or parameter does not exist or is not implemented
100	Invalid information element contents
101	The message is not compatible with the call state

Table D-4 Cause Code Descriptions (continued)

Cause Code	Description
102	The call was terminated when a timer expired and a recovery routine was executed to recover from the error
103	Parameter non-existent or not implemented - passed on (national use)
110	Message with unrecognized parameter discarded
111	Protocol error, unspecified
126	Call split. This is a Cisco-specific code. It is used when a call is terminated during a transfer operation because it was split off and terminated (was not part of the final transferred call). This can help determine which calls were terminated as part of a transfer operation.
127	Interworking, unspecified

Alarms

An alarm is issued when CDR or Diagnostic data is enabled, and the system is unable to write the data into the database.

Unable to write CDR data. (Alarm # 1711 - Major Alarm)

The system attempted to open the database, and was unsuccessful. Probable causes include:

- Cisco CallManager does not have sufficient privileges to open the file for writing to the database. Make sure Cisco CallManager has privileges that will permit write operations.
- The path is not set up, or the database server is down.