



MediaSense Terminology

This section identifies the commonly used MediaSense terms and provides a conceptual context for your reference and understanding.

- [Play back, page 1](#)
- [Blog recording, page 2](#)
- [Media forking, page 2](#)
- [Sessions and recording sessions, page 2](#)
- [Glossary of Common Terms, page 3](#)

Play back

You can search for a session and play the audio or video data for each session using the integrated Search and Play application or by using the MediaSense APIs. See the *Cisco MediaSense Developer Guide* (http://www.cisco.com/en/US/products/ps11389/products_programming_reference_guides_list.html) for more information about using the APIs.

You can play back MediaSense recordings using the Real Time Streaming Protocol (RTSP) or by downloading the recordings as .mp4 or .wav files.

- **Playback:** You can playback MediaSense recordings using the integrated Search and Play application or on any player which supports RTSP, .mp4, or .wav formats (for example, VLC—VideoLAN Client). If you listen to a forked media recording using VLC, you can only listen to one track at a time, and not both at the same time.
- **Download:** If you prefer to listen to both audio channels and view the video at the same time, export any MediaSense recording to .mp4 or .wav format using the **convertSession** API. This API returns the URL from which you can access the converted file. You can then download that file using standard HTTP access methods. Using a downloaded .mp4 or .wav file, you can listen to both audio channels and view the video at the same time.

Converting to .mp4 or .wav format also makes the file portable and allows you to copy it to a location of your choice.

- Client applications can communicate directly with the MediaSense media service by using the `downloadUrl` parameter in the Session Query APIs. Each API has a `downloadUrl` only for AUDIO tracks.

You cannot download MediaSense video tracks in the RAW format. The downloaded recording is available only in the RAW format.

This URL is conditionally present in the session query response only if the `sessionState` is `CLOSED_NORMAL` or in the `sessionEvent` only if the `eventAction` is `ENDED`. For other sessions in other states, (`ACTIVE`, `DELETED`, or `CLOSED_ERROR`), `downloadUrl` is not available. See the *Playing Back Recordings* section in the *Cisco MediaSense Developer Guide* (http://www.cisco.com/en/US/products/ps11389/products_programming_reference_guides_list.html) for more information.

Blog recording

MediaSense enables you to create blog recordings (audio and video) using supported Cisco IP Phones. After the recordings are made, third-party applications can publish them.

A blog recording is initiated in one of the following ways:

- By a user who dials into a MediaSense server.
- By the MediaSense server calling a user phone in response to an API request.



Note CUBE deployments do not support direct outbound recording.

Media forking

All Cisco IP phones that MediaSense supports have a built-in bridge (BIB) that allows incoming and outgoing media streams to be forked. MediaSense makes use of this capability to record inbound and outbound forked media. For more details about media forking, see the Unified CM documentation at http://www.cisco.com/en/US/products/sw/voicesw/ps556/prod_maintenance_guides_list.html.

CUBE does not have a BIB because the call forking is performed within the CUBE application—not from a phone.

Sessions and recording sessions

In MediaSense, a *session* is a recorded monolog, dialog, or conference that can involve one or more participants. A MediaSense session is the same as a *recording session* in Unified CM. See the *Cisco Unified Communications Manager Features and Services Guide* at http://www.cisco.com/en/US/partner/products/sw/voicesw/ps556/prod_maintenance_guides_list.html for more information about recording sessions.

The participants in a session use a *device* to participate in a MediaSense session.

A *device* is a physical entity that can be an endpoint or a personal computer and refers to any item that can be recorded. A device is identified by a `deviceRef` which is a phone number or extension for each device. The `deviceId` is the unique identifier for each device and it corresponds directly to the name of the device (like the MAC address or Universal Device Identifier—UDI).

A session can be *live* (active) or *recorded* (completed). A live session can be monitored and recorded at the same time. A recorded session can be played back at any time.

Glossary

Active server

An active server is a primary server or secondary server with one instance of the API service, configuration service, call control service, media service, database service, and the SM agent. A MediaSense cluster must have one or two active servers. Replication is available in both active servers. To ensure high availability, if one active server goes down, the other active server can handle the complete load for both servers.

API service

The application programming interface (API) service is a feature service. Each MediaSense cluster can only have two instances of the API service. One instance is in the primary server and another instance is in the secondary server. Each API service must have a corresponding configuration service. If a MediaSense cluster has more than two servers, the additional servers do not have an API service or configuration service. Each instance of the API service corresponds directly to one instance of the meta database.

Call control

MediaSense uses the session initiation protocol (SIP) to control new calls, transferred calls, and calls that are placed on hold.

Call control service

Call control service communicates with the network layer, media service, and API service to provide key recording functions for MediaSense. One instance of the call control service is present in each server in a cluster.

Cluster

MediaSense servers are deployed in a cluster. A cluster can contain from one to five servers. Each cluster can provide basic media recording, database storage, and scalable recording capacity.

Configuration database

The configuration database is often referred to as the 'config' database. It stores log level and trace mask information. Each instance of the configuration database corresponds directly to one instance of the configuration service. Although the configuration database is not directly exposed to end users, you can indirectly configure functions such as service activation in the MediaSense Serviceability web portal.

Configuration service

Configuration service is a feature service. Each instance corresponds directly to one instance of the configuration database. Each MediaSense cluster can only have two instances of the configuration service. One instance is in the primary server and the other instance is in the secondary server. When one configuration service does not function, data can continue to be written to the other configuration service because MediaSense uses a peer-to-peer database model.

Each configuration service on the primary server and secondary server must have a corresponding instance of an API service. If a MediaSense cluster has more than two servers, the additional servers do not have a configuration service or an API service.

Database

MediaSense has two databases: the configuration database and the meta database. The general term "database" is used to refer to both of them.

Database service

The database service controls the configuration database and the meta database. Each MediaSense cluster can only have two instances of the database service. One instance is in the primary server and the other instance is in the secondary server.

Device

A *device* is a physical entity such as an end point or a personal computer that can be used to make recordings. Each device is identified by a unique deviceRef or Device Ref.

Device reference

A device reference is called a deviceRef in the API service and a device ref in the administration service. It refers to the phone number, IP address, or the URI/URL of each device. One or more participants can be associated with multiple device references.

Diagnostics

MediaSense diagnostics is a network service. This service is present in all MediaSense servers for debugging and troubleshooting purposes.

Expansion server

A MediaSense deployment can have a maximum of three expansion servers. Each expansion server has one instance of the call control service and one instance of the media service. Expansion servers have no instances of the API service or the database service.

Feature service

Feature services enable you to configure and monitor all servers in a MediaSense cluster.

High availability

High availability means that if one server fails, the other server can handle the complete load for both servers in a MediaSense cluster. The data is load balanced between both servers and data replication is available in both servers.

Live (active) session

A live session is a call in progress and can be monitored and recorded at the same time. When it is finished, it becomes a recorded session that can be played back at any time.

Media service

Media service is a feature service. It terminates media streams for storage on a local disk. One instance of the media service is present in every server in a MediaSense cluster.

Media stream

A media stream refers to the packets going through an audio channel or video channel in a live or recorded session. It refers only to a live session. It does not refer to a recorded session. A recorded media stream is called a track.

Meta database

The meta database stores call history and metadata information associated with each recording. Each instance of the API service corresponds directly to one instance of the meta database.

Network services

Network services enable you to configure and monitor overall system functions. After you have installed MediaSense and re-booted your server, network services are enabled by default on all servers in a cluster.

Participant

A participant refers to people or end points involved in a session. Participants use a device to conduct a session. Participants are identified by a unique device reference, which is a phone number, IP address, or URL. During the same session, each track is associated with only one participant, the participant who is generating the media for that track. During different sessions, each track can have one or more participants.

Perfmon agent

This network service controls the performance monitoring infrastructure. It has no separate user interface and operates seamlessly within MediaSense serviceability administration.

Primary database

The configuration service in the first main server in any deployment is called the primary database. The configuration service in the second main server in any deployment is called the secondary database.

In a MediaSense cluster, configuration requests are sent to the primary database and the secondary database. If the primary database is functional, data is written to the primary database and then replicated to the secondary database. If the primary database is not functional, data is not written to ensure data integrity. If the primary database is not functional for a substantial period of time, you can manually promote the secondary database to be the new primary database so that data can be written to it. When the original primary database begins functioning again, it becomes the new secondary database.

Primary server

The primary server is the first server in the cluster. After you install MediaSense and re-boot the primary server, all MediaSense feature services are *enabled by default*.

Publisher

In MediaSense clusters, the primary and secondary servers are publishers (peer-to-peer).

Recorded (completed) session

A recorded session has been completed and can be played back at any time.

Recording types

MediaSense makes two types of recordings:

- Forked media recordings are made from IP phones. These recordings have two audio channels.
- Direct call recordings are made to and from MediaSense to any phone. These recordings have one audio channel and one optional video channel. (They are often called blog recordings in this document.)

Secondary database

The configuration database in the secondary server in a cluster is called the secondary database.

Secondary server

Each cluster can have only one secondary server. After you access the administration service and enable all feature services, you can assign that server as the secondary server. It is paired with a primary server to ensure high availability.

Session

A session is a recorded monologue call, dialog call, or conference call. A session is identified by a sessionID (or Session ID) and contains one or more tracks.

A MediaSense session has the same meaning as a recording session in Unified CM. See the *Cisco Unified Communications Manager Features and Services Guide* (http://www.cisco.com/en/US/partner/products/sw/voicesw/ps556/prod_maintenance_guides_list.html) for more information about its recording sessions.

Session ID

The unique identifier for a session.

SM agent

Storage management agent (SM agent) monitors the overall storage in each server in a cluster and generates threshold events based on disk usage. It is available in all servers in the cluster.

System service

This network service controls service operations. It does not have a separate user interface and operates seamlessly within the MediaSense administration service and MediaSense serviceability administration.

Tag

System-defined tags are brief, arbitrary text strings that associate individual sessions using the Web 2.0 APIs. MediaSense stores tags with each session. MediaSense uses tags to mark certain actions which occurred during the session (such as , pause and resume) or to mark when the media inactivity state changes (as reported by the SIP signaling). While most tags are associated only with a session, media inactivity state change tags are associated with a session and with a specific track in the session.

Track

A track identifies each media stream and quantifies it with additional data such as participants, duration, startDate, and trackNumber. Each track is specific to one audio stream or one video stream. Each track can be associated with multiple device references. Each session contains one or more tracks.

Track ID

The unique identifier for a track.