



Product overview

Cisco MediaSense is a SIP-based, network-level service that provides voice and video media recording capabilities for other network devices. Fully integrated into Cisco's Unified Communications architecture, MediaSense automatically captures and stores every Voice over IP (VoIP) conversation which traverses appropriately configured Unified Communications Manager IP phones or Cisco Unified Border Element (CUBE) devices. In addition, an IP phone user or SIP endpoint device may call the MediaSense system directly in order to leave a recording consisting of media generated only by that user. Such recordings can include video as well as audio—offering a simple and easy method for recording video blogs and podcasts.

Since forked media can be recorded from either a Cisco IP phone or a CUBE device, MediaSense allows you to record a conversation from different perspectives. Recordings forked by an IP phone are treated from the perspective of the phone itself—any media flowing to or from that phone gets recorded. If the call gets transferred to another phone however, the remainder of the conversation does not get recorded (unless the target phone has recording enabled as well). This perspective may work well for contact center supervisors whose focus is on a particular agent.

Recordings forked by CUBE are treated from the perspective of the caller. All media flowing to or from the caller gets recorded, no matter how many times the call gets transferred inside the enterprise. Even interactions between the caller and an Interactive Voice Response (IVR) system where no actual phone is involved will be recorded. The only part of the call which will not be recorded would be a consult call from one IP phone to another—for example, as part of a consult transfer. (Even that can be recorded if Unified Communications Manager is configured to route IP phone to IP phone calls through a CUBE.) This perspective may work well for dispute resolution or regulatory compliance purposes, where the focus is on the caller.

No matter how they are captured, recordings may be accessed in several ways. While a recording is still in progress, it can be streamed live ("monitored") through a computer which is equipped with a media player such as VLC or RealPlayer, or one provided by a partner or 3rd party. Once completed, recordings may be played back in the same way, or downloaded in raw form via HTTP. They may also be converted into .mp4 or .wav files and downloaded in that format. All access to recordings, either in progress or completed, is through web-friendly URIs. MediaSense also offers a web-based Search and Play application with a built-in media player. This allows authorized users to select individual calls to monitor, playback, or download directly from a supported web browser.

In addition to its primary media recording functionality, MediaSense offers two other capabilities.

It can play back specific video media files on demand on video phones or supported players. This capability supports Video in Queue (ViQ), Video on Demand (VoD), or Video on Hold (VoH) use cases in which a separate call controller invites MediaSense into an existing video call in order to play a previously designated recording. An administrator can upload studio-recorded videos in MP4 format and then configure individual incoming dialed numbers to automatically play those uploaded videos. The call controller plays the video by sending a SIP invitation to MediaSense at the dialed number.

MediaSense can also integrate with Cisco Unity Connection to provide video voice-mail greetings. Videos are recorded on MediaSense directly by Unity Connection subscribers and are then played back to their video-capable callers before they leave their messages.

Media recordings occupy a fair amount of disk space, so space management is a significant concern. MediaSense offers two modes of operation with respect to space management: retention priority and recording priority. These modes address two opposing and incompatible use cases; one where all recording sessions must be retained until explicitly deleted (even if it means new recording sessions cannot be captured) and one where older recording sessions can be deleted if necessary to make room for new ones. A sophisticated set of events and APIs is provided for client software to automatically control and manage disk space.

MediaSense also maintains a metadata database where information about all recordings is maintained. A comprehensive Web 2.0 API is provided that allows client equipment to query and search the metadata in various ways, to control recordings that are in progress, to stream or download recordings, to bulk-delete recordings that meet certain criteria, and to apply custom tags to individual recording sessions. A Symmetric Web Services (SWS) eventing capability enables server-based clients to be notified when recordings start and stop, when disk space usage exceeds thresholds, and when meta-information about individual recording sessions is updated. Clients may use these events to keep track of system activities and to trigger their own actions.

Taken together, these MediaSense capabilities target four basic use cases:

- 1 Recording of conversations for regulatory compliance purposes (compliance recording).
- 2 Capturing or forwarding media for transcription and speech analytics purposes.
- 3 Capturing of individual recordings for podcasting and blogging purposes (video blogging).
- 4 Playing back previously uploaded videos for ViQ, VoD, VoH, or video voice-mail greeting purposes.

Compliance recording may be required in any enterprise, but is of particular value in contact centers where all conversations conducted on designated agent phones or all calls from customers must be captured and retained and where supervisors need an easy way to find, monitor, and play conversations for auditing, training, or dispute resolution purposes. Speech analytics engines are well served by the fact that MediaSense maintains the two sides of a conversation as separate tracks and provides access to each track individually, greatly simplifying the analytics engine need to identify who is saying what.