BULK RETRIEVAL TECHNIQUES FOR VOICE LOGGERS

Andrew Stevens, Ph.D.
May 2025

Executive Summary

Many businesses and agencies record their telephone traffic for the purposes of record keeping, compliance, and data analysis. Voice loggers are systems specifically designed for this purpose and are used to capture every telephone call and also provide the ability to replay recordings at a later date. However, there are instances where the default one-at-a-time manual playback capability of the recorder is insufficient.

Bulk retrieval of archived recordings is often necessary in situations such as migration to a new logger platform, ingestion by speech recognition tools or AI engines, and litigation discovery. But while all voice loggers are specifically designed for efficient simultaneous recording of new calls, few loggers are also designed for efficient export. And bulk exporting can be extremely complex due to the presence of incompatible storage formats, a multiplicity of speech encoding algorithms, and disparity among metadata field sets.

This paper addresses the intricacies of extracting audio and metadata from voice loggers while maintaining data integrity and completeness. Topics covered include: estimating the total amount of stored data, interpretation of per-call metadata, conversion of recorder archives to standard file formats, requirements for target repositories to hold the converted audio and metadata, and adherence to rules and regulations for compliance and legal attestation. We will specifically discuss voice loggers which store audio in a digital data format (comprising most recorders since 1990).

VOICE LOGGER FUNDAMENTALS

Functionality

Voice logging is the act of recording telephone and other audio traffic for the purpose of creating an unambiguous record of conversations, including not just the words spoken but the specific dates and times of calls and the parties involved. It is often used for regulatory compliance, quality assurance, or public safety purposes. Typical applications include financial trading floors (to verify buy and sell orders or detect improper trading), telephone call centers (to study agent effectiveness and customer satisfaction), and emergency hotlines (to monitor response times, provide evidence for legal actions, or develop statistical data around crimes).

Voice loggers are the systems used to create and store the recordings as well as their associated timestamps, phone numbers, call durations, and other related fields (collectively referred to as the “per-call metadata”). Voice loggers might be physical hardware located in a telephone closet or data center, or may be virtual hardware located either on-premise or in the cloud. They may be connected to individual telephone extensions, open microphones (“hoot-n-holler boxes”) or PBX trunk lines, and they record all calls and metadata as they occur. Voice logger platforms are designed to support a specified number of “seats” corresponding to call center agents, or "channels" corresponding to the maximum number of users or extensions which may be recorded simultaneously. A voice logger must have an accurate clock (to record timestamps), a storage archive (to save the recorded audio), and a database (to store metadata). A voice logger may also have the ability to detect dialed digits within a phone call (DTMF tones) or sense minimum levels of speech activity so that it can pause recording during long periods of silence.

Some manufacturers of voice loggers over the past 35 years include ASC, Calabrio, CyberTech, Dictaphone, Eyretel, Genesys, Mercom, NICE, OAISYS, Racal, Red Box, Verint, and Witness.

Can voice loggers perform bulk data retrieval?

Voice loggers are typically capable of exporting calls only in small quantities, perhaps just a few dozen at a time. This suffices in many common scenarios such as the disambiguation of details around a particular stock trade. Yet there are many cases where retrieval of 10,000+ calls is required. For example, we have seen legal investigations into rogue equity traders that involve multiple custodians for many months or years. Or companies that wish to upgrade or consolidate their multiple recorder platforms might need to bulk export millions or even billions of calls. If performed on the original system, these retrievals would entail a laborious process via repeated queries through the logger's native interface. Some recorders provide an API for automated export of recordings and metadata, but these are frequently unavailable. Most voice loggers demonstrate a large functionality gap when it comes to bulk export.

Storage and Retention of Voice Data

Voice loggers generate a large amount of data and archive to a variety of mass storage devices. Older units used physical media such as magnetic tapes or optical disks. Modern voice loggers use hard drives, network file shares, network-attached storage (NAS), enterprise data archives (e.g. EMC Centera), or cloud storage (e.g. AWS S3 or Azure Blob Storage). In most cases, the archived files or media are fundamentally different from conventional hi-fi audio recording formats (e.g. audio CDs, cassette tapes, or MP3 files). Most logger manufacturers use their own digital audio format, and the recordings are designed to be replayed only within the original system.

Multiple reasons mandate that the recorded audio be retained for a specific interval (the "retention period"). For telephone contact centers, the retention might be several months for quality assurance purposes; for financial services firms, compliance regulations might dictate a three to seven year retention; for life insurance companies, retention could extend for the duration of the policy (perhaps decades); for emergency hotlines or air traffic control towers, permanent retention might be indicated in case the recordings are needed to investigate a murder or plane crash. In addition, retention periods often vary among countries and jurisdictions.

Voice Data Retrieval and Replay

Capturing and storing voice data is only half of the story. Voice loggers must also be able to retrieve files from storage and replay calls. The ability to apply specific search criteria to the retrieval (such as caller ID, agent name, and date range) is fundamentally important. This functionality is usually provided via a GUI ("graphical user interface"), often consisting of a browser-based interactive search form which generates a list of matching recordings. Clicking on any single match causes the corresponding audio to play in the browser or optionally allows download of the audio file in WAV or MP3 format, with an accompanying CSV file for the associated metadata.

Download of multiple matching results in a single step (e.g. by checking boxes in the result set) might be available but is practically limited by large file sizes and available network bandwidth. Thus, any wholesale retrieval via the GUI must be broken up into multiple smaller downloads, with the added risk of human error due to the many manual operations necessary for each search and download.

AUDIO METADATA

What Information Does the Metadata Hold?

All voice loggers store metadata related to each recording, always minimally including the call start time, stop time, and a channel identifier. Depending on the generation and the sophistication of the voice logger, there may only be a handful of additional fields such as agent name and call direction; in other cases hundreds of fields might be stored in a relational database including extension number, extension type (e.g. handset or microphone), trunk name, dialed digits (aka DNIS or "dialed number identification service), ANI (automatic number identification, aka Caller ID), and more. Some loggers also support installation-specific custom fields which might contain business-related information like invoice number or customer name. The availability of these fields depends on the capabilities of the voice logging platform, its ability to connect to the PBX via CTI (computer-telephony integration), possible integration with CRM (customer relationship management) software, and the proper setup and ongoing maintenance of the integrations.

Associating Metadata to Audio Recordings

Depending on the logger model, the association of the metadata to actual recordings might be "implicit" because they are stored together in the same file, or the association might require a matching operation because the extended metadata are stored separately from the audio, e.g. in a separate relational database. Storing the metadata in an external database is convenient and results in a more modular system architecture and more sophisticated reporting. However, it can also lead to problems for retrieval if both subsystems are not backed up properly and synchronized. We have seen cases where the database has failed independently of the recorder, resulting in thousands (or even millions) of recordings with missing metadata because nobody noticed the database failure. In other cases, the logger is retired and the archived audio is safely stored, but nobody remembered to backup the database. In these cases, those audio files with no corresponding database entry have lost much of their value.

Time & Duration Discrepancies between Audio Recordings & Metadata

The method for determining the beginning and end of an individual call can vary depending on the particular logger platform, the PBX model, and the method of their interconnection. Even if the mechanism for matching metadata and audio is straightforward, the interpretation of the combined metadata sets can be confusing, as described below.

Generally speaking, a telephone "call" is considered to begin when the calling party picks up their handset (or colloquially speaking, "goes offhook"), and ends when one of the parties hangs up (or "goes onhook"). A "call detail record" (CDR) for each individual call is stored in a database inside the PBX. Because the offhook/onhook events are only detectable by the PBX, the corresponding CDRs are only available to the voice logger if it is integrated with the PBX via a CTI connection, thus allowing offhook events to be shared among the two systems.

In cases where the logger is not integrated with the PBX (and also does not use VoIP), then the only way the logger can detect offhook/onhook is by watching for activity on the monitored voice circuits, also called “VOX” detection. But there are multiple scenarios in which problems may arise. For example, when a call is placed on hold, the resulting silence may fool the logger into thinking that the call has ended and thus cause it to stop recording. When the call resumes, the logger starts recording again, but has no way of knowing whether the new activity is a new call or the resumption of the previous call. Thus, a single call (as determined by the PBX) might be stored as several distinct recordings or “segments” (as determined by the voice logger). Alternatively, a caller might hang up a call and then immediately place a new call, with only a short period of silence between calls. In this case, the logger does not always detect the hangup and may combine the two calls (as determined by the PBX) into a single segment (as determined by the voice logger). In another scenario, trading floors often employ open microphones (“hoot-n-holler” boxes) which are connected to the PBX and record all activity, 24 hours per day. From the PBX’s perspective, the open microphones are treated identically to ordinary handset calls, but can have durations of days (or even months) per call. On the other hand, voice loggers usually have a maximum segment length, typically in the vicinity of three to twelve hours. Thus, a single open-microphone call in the PBX (perhaps having a duration of 4 weeks) can correspond to hundreds of segments in the voice logger.

The upshot is that some voice logger installations have two distinct sets of metadata: (a) the segment metadata as determined by VOX detection, and the PBX metadata as determined by offhook/onhook events. Consequently, confusion can arise during the review of the exported recordings due to the apparent mismatch of the two metadata sets. Some solutions to the confusion are to (a) “stitch” the calls back together from their individually-recorded segments, with the penalty of requiring additional disk space for any reinserted silence; (b) “virtually” stitch the segments and silence via a playback GUI (though no such aftermarket product currently exists that we are aware of), or (c) simply educate the reviewers about the intricacies of the two metadata sets.

Many modern voice logger installations that use VoIP transmission do not suffer from VOX/PBX metadata mismatch. This is because VoIP packets include embedded offhook/onhook event data, thus avoiding the need for VOX detection and allowing perfect synchronization between the PBX and the logger.

HOW MUCH DATA IS IN MY VOICE LOGGER SYSTEM?

Voice loggers generate huge amounts of digitized audio data. When performing bulk retrievals, the amount of necessary effort is usually proportional to the amount of data residing in the recorder. But how does one measure the total amount of data, and in what units? Several possible measurement criteria include total recorded hours, total number of recordings, total disk space, or total file count in the archive

Unfortunately, when taken individually, these quantities rarely predict the necessary effort or complexity of the retrieval with any accuracy. For example, an archive that contains 1 million recorded hours might contain 1 million one-hour recordings or 360 million ten-second recordings. The corresponding difference in effort to extract the audio and per-call metadata in these two cases is substantial. The complicated relationship between these quantitative parameters is described below.

Archive Storage Space and Codecs

An express design goal for most voice loggers is to minimize as much as possible the required storage space for the archived audio. Audio compression (or “encoding”) is used to reduce the number of bytes necessary to store the audio, and the mathematical algorithm used to implement a particular encoding is called a codec.

An abundance of codecs exist for compressing digital speech, including published standards (G.711, G.722, G.723.1, G.726, G.729, GSM 06.10, MP3), open source projects (Speex, Opus), licensed algorithms (TrueSpeech, IMBE), and proprietary algorithms (NICE ACA, Verint SBC). These codecs all have conflicting tradeoffs between voice quality, compression ratio, computational complexity, silence suppression, multichannel support, and playback compatibility. As an example: a 1 terabyte archive might contain 5 million recordings encoded with stereo G.711 compression, or 260 million recordings encoded with mono AMBE compression. Some common voice logger codecs and corresponding characteristics are listed in Table 1.

How Many Hours of Audio Am I Storing?

Determining the number of recorded hours present within a set amount of storage is not straightforward. It depends on several key factors: the employed codec, the optional use of stereo recording, and the possible suppression of silence when storing the recordings.

- **Codecs and Storage.** The storage efficiency for a given codec is indicated by its stated bitrate which corresponds to the amount of disk space necessary to store one second of audio. For example, G.711 encodes audio at 64000 bits per second (“bps”) and so each recorded second requires 64000 bits (or 8000 bytes or 8 kilobytes) for storage. Thus, a 1 hour G.711 recording requires 28.8 megabytes of space (64000 bits/sec × 60 sec/min × 60 min/hour ÷ 8 bits/byte). For the more modern G.729 codec, each recorded second consumes 8000 bits (or 1000 bytes), an 8× improvement over G.711, and thus the same 1-hour recording

What is a codec?

Digital audio recording always starts with converting the analog signal from the microphone into a stream of binary numbers (“analog-to-digital conversion”). Then, a codec (“coder-decoder”) may be applied in order to compress the audio bytes while retaining fidelity and dynamic range. A codec is simply a mathematical algorithm which operates on the microphone bytes, and “speech codecs” are optimized by taking advantage of specific human voice patterns. For example, most speech codecs account for the fact that the human voice uses less audio bandwidth (about 8 kHz) than the full range of human hearing (about 20 kHz, also called “hi-fi”). And as it turns out, even the full 8 kHz is largely unnecessary for speech to remain intelligible, and so most speech codecs discard all frequencies above 3.4 kHz. As a result, digital “voiceband” signals can be transmitted (or stored) with far fewer bytes than hi-fi audio.

The oldest speech codec is G.711 and uses a logarithmic transform to compress 14-bit audio into 8 bits, while also causing an inaudible loss of quality for loud voices. Linear predictive encoding (“LPC”) is another numerical technique (used by the GSM 06.10 codec) which uses a mathematical model of the human vocal cords, throat, and mouth to minimize the transmission of sounds which simply cannot be formed by human physiology. Code-excited linear prediction (“CELP”) provides even more efficient compression, at the expense of additional computing power necessary to implement its complex algorithm (used by G.729 and G.723.1 codecs). Some codecs support higher bandwidth (“HD Voice” supported by G.722). Research on new codecs is ongoing, and the current state-of-the-art codec (as of 2025) is Opus which provides excellent compression and voice quality but unfortunately is not yet supported by all computer operating systems. Note that codecs are also used for other types of data, e.g. digital images often use the JPEG algorithm to compress digital photos from “raw” format to a much smaller .JPG file.

consumes only 3.6 megabytes. On the other hand, G.729 requires a much more powerful CPU than G.711 in order to implement its complex mathematical algorithm. In addition, its playback quality is not quite as good as G.711 because its compression algorithm can slightly distort the audio upon playback (though largely indistinguishable to ordinary human ears).

- **Stereo vs. Mono and Storage.** Stereo recording is an optional feature supported by many recorders and is often used in contact centers. When enabled, the external caller is recorded onto one channel and the call center agent is recorded onto the other. Advantages to stereo recording include better intelligibility, the ability to determine absolutely which party spoke which words, and improved analytics capability. However, the corresponding stereo recordings will often have twice the size as mono recordings.

- **Silence Suppression and Storage.** Some loggers employ silence suppression to avoid wasting disk space during lengthy silences. For example, if a call uses the G.729 codec and contains 30 seconds of silence, the logger might store a brief message indicating the silence duration (perhaps consisting of only a few bytes) instead of writing the entire 30000 bytes required to store the 30 seconds of silence. On many stock trading floors, turret microphones often record activity 24 hours per day, even during off hours. In these instances, silence suppression can allow a substantial reduction in necessary storage space by simply not recording nighttime silence.

- **Metadata and Storage.** In systems with an external relational database, it is possible in theory to query the database to extract the total number of recorded hours. However, in practice, the database does not always accurately reflect the total amount of archived audio. For example, faulty retention policies in the logger can sometimes cause audio to be deleted without corresponding metadata deletion, or vice versa. This can cause the database to become out of sync with file storage, thus causing the database to contain an incorrect tally of hours or recordings. In another example, a temporary database outage might cause metadata to be completely lost while audio continues to be captured and written to the archive. Further complicating the issue, the actual mechanism for correctly querying the relational database tables for call count and duration is not always readily apparent.

- **Average call duration (“ACD”) and daily traffic totals.** Many contact centers carefully calculate ACD and daily traffic as a measure of agent effectiveness and efficiency. However, these metrics often fail to account for multiple recording segments per call (due to holds or transfers), failed connections (which might get excluded from the ACD calculation), or other exclusions.

In most cases, we find that the best metric to indicate the complexity of a bulk extraction is usually “recording count,” which is defined as the total quantity of stored audio segments and corresponding start and stop timestamps. For archives which store in a one-file-per-recording file format (e.g. NICE NMF, Genesys SASF, or Oaisys PVD files), the recording count is simply the number of stored files. For archives which write audio in a many-recording-per-file format (e.g. Verint TAR or Red Box FRAME files), an accurate inventory sometimes necessitates inspection of the actual archive files and can be difficult to obtain a priori. In these cases, a recording count estimate can only be obtained via database query or disk space estimate. For older recorders where the archive is stored to external media (magnetic tape or DVD-RAM), the best metric is the total media count for each storage format (e.g. DDS-4, AIT-2, double-sided DVD-RAM).

Table 1. Common codecs used for voice recording.

CODEC	BITRATE (bits/sec)	AUDIO BANDWIDTH (Hz)	YEAR INTRODUCED	REMARKS	SUPPORTED REPLAY ENVIRONMENTS
G.711 μ-law / A-law	64000	3400	1962	“Toll quality” voice, still in broad use as of 2025	All modern operating systems and HTML5
ADPCM / G.726	16000	3400	1973		
G.722	64000	7000	1988	HD Voice	
MP3	variable	variable	1991	Intended for hi-fi audio	All modern operating systems and HTML5
GSM 06.10	13000	3400	1992		All modern operating systems (mono only)
G.723.1	5336	3400	1996		
TrueSpeech	8500	3400	1996		
G.729	8000	3400	1996		
AMBE / IMBE	2400	3400	1997	Common for two-way radio	
NICE ACA	5600	3400	1998		
Speex	variable	3400	2003		
Opus	variable	variable	2012		HTML5

THE NEED FOR BULK EXPORT

With a deeper understanding of voice logger systems, the data they generate, and how that data is stored, we can turn to the challenge of bulk export. As stated earlier, bulk retrieval is often necessary due to:

- **Compliance.** Regulatory compliance dictates that data be retained and that it be quickly retrievable based on specific criteria such as date interval and agent name. These search criteria will often match a huge quantity of records (thousands or more) which is impractical to perform using the limited playback mechanism which is native to most voice logger systems.

- **Litigation support.** Another common reason for bulk retrieval is for trial discovery. For example, an insider trading investigation might take years to come to light and might require the production of all recordings for a given stock trader over for an interval of months or years. The corresponding recordings – often numbering in excess of 10,000 recordings for each trader – need to be retrieved quickly and accurately, must be of a digital format acceptable to all parties and the courts, and must be technically defensible in a court of law in terms of their integrity.

- **Voice Logger migration.** When organizations purchase a new voice logger system, they often still need the ability to replay the recordings from the previous system. In order to achieve this, one possibility is to simply retain the old system for the duration of the retention period (which might be seven years or more). Some vendors offer a reduced-cost "playback only" license for their platform, though this option can still be expensive due to ongoing annual support and hosting cost. And in some cases, this option is not possible because the old recorder is EOL ("end of life"). And although many (but not all) new recorders have a tool for importing external audio and metadata, many of the older recorders lack any such mechanism for exporting it. As mentioned earlier, many voice loggers have no bulk export capability, or else require special licensing and/or professional services charges from the recorder vendor to perform the export. We speculate that the reason for the poor availability of export tools is for the express reason of discouraging customers from switching vendors.

Why migrate your voice logger system?

Many reasons may drive organizations to upgrade their voice logger, including:

- **New features and analytics.** Constantly evolving technologies and processes such as workforce optimization, customer personalization, or AI-driven analysis are driving businesses to upgrade their recorders.
- **To support cloud hosting.** Large-scale corporate IT initiatives such as digital transformation or cloud migration require modernizing the call logging platform to better integrate with other business systems or hosting platforms.
- **To standardize on one recording platform.** Companies often find themselves with multiple voice logger systems as a result of corporate mergers and acquisitions. For cost, management, and compliance oversight purposes, they will often seek to standardize on one platform.
- **The current product is being discontinued.** In some cases, the logger manufacturer been acquired and their product subsequently declared "end of life" by the acquirer – as evidenced by the disappearance of many vendors over the years including Dictaphone, Eyretel, Racal, and VPI. In other cases, the manufacturer discontinues the logger in favor of a new system built using more modern software elements and techniques (examples include NICE NTR and Genesys PureConnect).

- **Speech Analytics.** Statistical analysis of large data sets (including but not limited to voice data) is an ongoing research area and has been known by many names over the years including data mining, machine learning, business intelligence, big data, and (most recently) artificial intelligence. Many of these sophisticated computational engines operate on textual data, thus the analysis of voice recordings is heavily dependent the successful conversion of voice "phonemes" into corresponding text. "Speech-to-text" technology continues to improve as computers get faster and the algorithms become more sophisticated. Major objectives include multi-language support (including dialects and accents), speaker differentiation, and rejection of background noise. Common applications of AI for telephone contact centers include measuring agent effectiveness and gauging customer satisfaction. For litigation support, AI may be used to analyze recordings in order to find specific keywords or custodians. Many voice logger manufacturers integrate their own (or a partner's) analytics platform which is able to ingest that particular manufacturer's archive format. But any organization that wishes to try alternative AI vendors must somehow bulk export the audio into an open format that is understood by the particular analytics engine (e.g. WAV files with G.711 compression).

A GUIDE TO BULK EXTRACTION OF VOICE LOGGER DATA

We will now address the challenges faced by voice logger owners who want or need to produce bulk audio data from their voice logger archives. We will also describe the advanced techniques developed by XOVOX which allow bulk retrieval beyond the extremely limited capabilities of the original loggers.

Choosing a Target Audio Format

When extracting audio from voice loggers, the choice of the target delivery file format depends on several criteria. A fundamental goal is that the audio should be converted into a file format which will replay in the widest variety of environments without requiring the installation of any specialized playback software (summarized in Table 2). Although the codecs in Table 1 are all commonly used to store voice logger audio, only a handful are appropriate conversion targets that will reliably replay across common computing environments (as indicated in the rightmost column in Table 1). Hence, transcoding of the original audio is often necessary in order to ensure replay compatibility.

The appropriate choice of target file format also depends on the conflicting criteria of audio quality and storage space (see Table 3). Ideally, the audio quality of the converted files should remain comparable to the original recording and not be audibly degraded. However, sometimes a slight degradation in the audio is acceptable if it results in a substantial reduction in storage space.

If the goal is the highest audio quality and the broadest replay compatibility, then WAV file format with G.711 encoding is often the best choice. This format is guaranteed to not degrade audio upon transcoding from most speech codecs (the rare exception being "HD Voice" codecs like G.722), it supports stereo, and it replays in every known environment. The high audio quality also makes this format the preferred choice for ingestion into speech analytics or AI engines.

Table 2. Common audio playback software for voice recordings

PLATFORM	AUDIO PLAYER
Microsoft Windows	Windows Media Player
Apple MacOS	QuickTime Player
Apple iOS	Built-in iOS audio player
Android	YouTube Music, Google Play
HTML5 Web Browser (Chrome, Edge, Safari)	Built-in browser-based audio player

Table 3. Common target audio file formats for voice logger extractions

FILE FORMAT	PROS / CONS
WAV file with G.711 (.wav)	Highest quality for voiceband audio, replays in all environments. High storage usage. Recommended where the target is audio ingestion into analytics/AI engines, or for legal cases.
WAV file with GSM 06.10 (.wav)	Good quality audio, low storage usage. Only supports mono audio, does not replay in HTML5 Web Browser. Recommended where the goal is long-term storage for regulatory compliance.
MP3 (.mp3)	Replays in all environments, low storage usage. Originally intended for hi-fi and thus requires extreme care when encoding voice audio to avoid quality issues. Recommended format in cases where the original recorder archives in MP3 format (thus avoiding need for transcode).
Opus (.opus)	Optimized for speech and hi-fi with low storage usage. As of 2025, only replays in HTML5 Web Browsers and some operating systems.

Alternatively, if the goal is to minimize the consumed storage space, then WAV file format with GSM 06.10 encoding is a frequent choice. This format introduces a slight audio degradation upon transcoding, but provides a 5x reduction in storage space vs. WAV with G.711 (or 10x reduction if also converting from stereo to mono). Replay compatibility is also broad (except for HTML5 Web browsers). This is the format of choice in cases where recordings are rarely replayed and the main goal is to observe compliance regulations for retention.

What's the best codec to ensure universal playback of my voice recordings?

We find that the codecs which have the broadest compatibility amongst today's common playback platforms (Windows, MacOS, iOS, and Android) are GSM 06.10 for mono recordings (13000 bps) and G.711 for stereo (128000 bps). For recorders that natively store to MP3 file format, we recommend keeping the recordings in MP3. Opus encoding is an option in cases where playback solely via an HTML5 Web browser is preferred.

MP3 is often suggested as a target delivery format due to its popular use for compressing hi-fi audio. However, MP3 is not optimized for telephone audio and will generate files approximately the same size as WAV with GSM 06.10. On the other hand, MP3 supports stereo, and has broad replay support.

The Opus codec is a relative newcomer (released in 2012) and it is optimized for both speech and hi-fi audio. It offers high audio quality and a low storage requirement. However, while it replays in HTML5 Web browsers, it is not guaranteed to replay in all OS environments (e.g. when audio files are delivered as email attachments). As of 2025, the broad acceptance of Opus is still evolving.

Note that converting to any codec with higher quality than G.711 (e.g. 16-bit linear PCM) serves no purpose because most modern voice traffic inevitably gets converted to G.711 (or other high-compression encodings) at some point during its end-to-end journey between calling parties. (Analogously, once a high-resolution digital photo has been reduced to low resolution, the photo cannot be converted back to high resolution.)

In some cases, the target for the retrieved recordings is not the human ear but rather a speech analytical engine. For example, some companies want to analyze their recordings to identify sensitive information such as personally identifiable information (PII) or personal health information (PHI) in order to prevent playback by unauthorized personnel ("redaction"). In these cases, the preferred codec is G.711 in order to obtain the best possible interpretation of the digitized audio.

Selecting a Target Audio Repository

Depending on the size of a given retrieval project, there are many available delivery options for the produced metadata and WAV files for subsequent search and replay.

For smaller jobs of less than one million recordings, spreadsheets (e.g. Microsoft Excel) are a simple and familiar format for the metadata. In this case, the spreadsheet can be easily searched for interesting calls (based on date, channel, etc.) and then the corresponding WAV files can be replayed via hyperlinks in the spreadsheet. The entire retrieval (spreadsheets plus audio) can usually be written to a flash drive or portable disk.

For larger projects (greater than one million recordings), several options exist for delivery with varying levels of cost and complexity.

- Some companies may have existing digital asset repositories (either hosted in-house or at a vendor). Metadata and audio delivery can be customized to meet the import requirements of the repository (typically XML, JSON, or CSV for metadata, and WAV for audio).

- In other cases, the customer is migrating logger platforms and would like the legacy audio to be imported into their new logger environment. This is achievable with varying levels of difficulty ranging from extremely simple (i.e. with full cooperation by the new logger vendor) to extremely complex (i.e. without any vendor cooperation and with the added risk of voiding the warranty on the new logger).

- A third option is the installation of a new audio repository which is specifically tailored for voice logger metadata search and replay. We have worked with several partners to deliver such repositories (both on-premise and cloud-based) which consist of a relational database and a Web-based server application to provide browser-based search and retrieval capabilities. Advantages include support for multiple simultaneous users, the ability to ingest data from multiple source loggers, and the avoidance of any application-specific software installation on the client device.

Retrieval Methodology

There are several approaches to retrieving voice logger audio, including: (a) using the original logger equipment; (b) using a bulk retrieval tool supplied by the original manufacturer, assuming such a tool exists; (c) writing a custom tool to query the original equipment via an existing API (which may or may not be documented or approved by the manufacturer); or (d) extracting the audio and metadata directly from the archive media or files, without use of any of the original logger hardware or software.

- As already detailed, retrieval via the original logger is slow and inefficient. Many loggers are designed to only provide one-at-a-time playback, and for some of the oldest (non-GUI) systems, playback may only be available through a speaker on the front panel of the hardware. Bulk retrieval is manual, error-prone, and may take months or years. Some systems allow retrieval of up to 100 recordings at a time via a GUI application. In this case, it is possible to employ multiple operators at multiple workstations in order to extract thousands of calls per day. However, this technique is vulnerable to quality problems due to its dependency on human keystrokes and mouse clicks in order to save each group of 100 retrieved calls.

- Some older loggers support client-server communication over a network (e.g. NICE 8.9 or Racal Wordnet), and it is possible in theory to write a program which emulates the client software and performs repeated retrieval requests without human intervention. However, such a program is still limited by the retrieval speed of the original system. Newer loggers (including all cloud-based platforms) provide a published specification for automated extraction via network connection, typically via a “RESTful API” (Representational State Transfer API). Automating the retrieval requires programming expertise by the customer and might be bandlimited by the platform (perhaps limited to one retrieval per second, implying that retrieval of 10 million recordings would take four months).

- Some manufacturers provide “playback only” environments (e.g. Eyretel eWare and NICE Playback Portal) or “extraction toolkits” (e.g. NICE ETK). However, these solutions do not always solve the issue of closed file formats, and are often expensive and slow.

At XOVOX, we have attempted all of these approaches and have determined that the fastest and most reliable approach for bulk retrieval is to extract the audio files directly from the archive media, without use of the original equipment and/or software. While the difficulty of this task varies depending on the logger manufacturer, the advantages in accuracy and speed can be enormous, oftentimes by as much as ten times faster. And by using generic equipment, the retrieval process can be scaled up by simply adding more servers and/or media readers to the job. Retrievals vary from logger to logger but generally follow this procedure:

Step 1: **Assign/deploy a computing environment.** For the one-time extraction of legacy recorder data, a temporary farm of Linux servers and fast disk storage should be allocated for the life of the project. These hosts can be deployed as physical servers or virtual machines, either on-premise, remotely, or in the cloud, with an appropriate method for login access such as VPN. For projects that involve ongoing extraction of new recordings, a permanent (but smaller) server environment must be allocated. For projects that involve physical media, the appropriate tape or optical readers must be installed in the servers.

Step 2: **Inventory** the logger archive. This involves generating a list of all of the stored files (or tapes or DVDs) in the archive. For loggers that use a relational database, also fetch all database tables, either from backup files or via read-only network connectivity to the database.

Step 3: **Fetch the source data**. For file-based archives, fetch a local copy of each archived file for subsequent processing. For loggers that archive to tape or DVD, perform a byte-for-byte image extraction from all media, thus allowing a complete fetch of the data in a single pass and also allowing immediate identification of any damaged or blank media.

Step 4: **Extract implicit metadata** from the raw data and reconstruct into individual call records, one record per recording. Depending on the logger, the available metadata fields may be minimal (limited to call start/stop times and channel number), or may be rich (including agent name, dialed digits, caller ID). For focused retrievals (e.g. limited to specific dates or channels), use the metadata to determine which files should undergo audio retrieval.

Step 5: **Extract audio** bytes from archive files. Demultiplex interleaved audio and reinsert suppressed silence as necessary to reconstruct into individual recording segments.

Step 6: **Transcode** the audio as necessary from high compression codecs (as listed in Table 1) into an open format (as listed in Table 3). For example, convert recordings stored with G.729 compression into WAV files with GSM 06.10 compression (or other file format) as needed for import into the destination system.

Step 7: **Extract extended metadata** from relational databases (if present). Match the metadata to the audio based on identifiers in the database or by analyzing timestamps and channel identifiers.

Step 8: **Convert all extracted metadata** (both implicit and extended) into the appropriate format as needed for import into the destination system. Common delivery formats include CSV, XML. JSON, and other text formats.

How reliable are tapes or DVDs for long-term storage?

In projects involving audio extraction from physical media, we sometimes see as many as 10% of the media with some kind of read error. Typical issues include snapped tapes, bad spots on tapes, scratched disks, and corrupted data. In most cases, data recovery techniques allow for successful extraction of 99% of the audio.

Step 9: **Reconcile** the extracted audio with the original logger archive. For example, if the original logger contains exactly 10 million recordings, then the extracted audio should identically contain exactly 10 million recordings (and not 9,999,999 recordings). Any discrepancies are accounted for in an exception list. Some common reasons for discrepancies include zero-length or truncated files written by the original recorder, accidental deletions in the original archive, failed backup/restore operations, damaged magnetic or optical media, temporary network outages on the original recorder, and a multiplicity of other possible failure modes.

Step 10: **Deliver** the extracted audio and metadata to an appropriate destination location for subsequent import into the target audio repository, e-discovery review platform, speech analytics system, or modern voice logger system. Common delivery destinations include hard disk, NAS, portable drives, or cloud storage. Note that copying millions of small files can sometimes take days or weeks, and many common file transfer protocols are unreliable at that scale. Wholesale deliveries involving millions of files require verification to confirm that all of the data actually arrived at the final destination.

Note that the above procedure is adjusted depending on requirements of the particular retrieval, and many of the steps can be completed in parallel. For example, if the recorder is not actively recording, then the audio and metadata steps can proceed simultaneously. As the extraction proceeds, the completion time can be extrapolated and additional servers may be added in order to accelerate the projected completion date.

If the source recorder is planned for shutdown but still in service, then the extraction is typically split into two parts: an initial migration of perhaps 95% of the data, and a second "delta" migration (after shutdown) for the remaining 5%. Alternatively, live sites may only need to retrieve each day's recordings (perhaps numbering a few thousand) for subsequent ingestion into a voice analytics engine; in this case, a small networked server appliance (installed locally or cloud-based) can perform all of the steps on a daily (or even hourly) basis, including fetching of the recordings from the file system and fetching of metadata via network connection to the live logger database.

FINAL NOTES ON THE LOGISTICS OF VOICE LOGGER BULK RETRIEVAL

Voice loggers are often installed in order to satisfy regulatory compliance rules, so the same rules will often limit the way retrievals can be performed. Compliance agencies and standards such as the SEC (USA), HIPAA (US), GDPR (EU), MiFID (EU), FCA (UK), FSC (South Korea), NAFR (China), ASIC (Australia), and PCI DSS (global) have strict rules relating to data security, data retention, and the transportation of backup media. Depending on the nature and jurisdiction of the recordings, sometimes the voice logger recordings must remain on premises; in other cases, they cannot leave the country, or can only be uploaded to cloud provider with a local data center. In cases of physical media, sometimes the original tapes cannot leave the premises but binary copies can be shipped for retrieval, or uploaded to cloud storage; in some cases, the media can only be shipped by "white glove" courier.

Overall, there is a surprising degree of complexity to this task of the bulk extraction and conversion of audio from voice loggers. These challenges arise from a combination of: the inherent, complicated technical architecture of the original logger hardware; the stringent limitations of various governing bodies regarding storage and retention; and the competitive and short-lived nature of voice logger manufacturers (as evidenced by the disappearance of many vendors including CyberTech, Dictaphone, Eyretel, Interactive Intelligence, Mercom, Racal, VPI, and Witness). In this report, we have described the many challenges related to the bulk retrieval of audio and metadata from voice loggers, how to prepare for such an endeavor, and the best approaches for a successful retrieval.

ABOUT THE AUTHOR & XOVOX

Andrew Stevens, Founder and President of XOVOX, has been working with all types of telecom equipment ever since he was a telephone repairman as an undergraduate at MIT. After receiving his bachelor’s degree in 1986 in electrical engineering, he worked at Bell Laboratories and IBM, and in 1995, he received his Ph.D. from Columbia University. In 2001, he founded XOVOX (formerly Electrical Science), a company specializing in finding elegant solutions to difficult and esoteric problems.

Today, XOVOX is the leading supplier of voice logger retrieval services for litigation support, trading floors, telephone call centers, and public safety hotlines. We have performed successful retrieval projects on six continents worldwide, totaling billions of recordings over decades, covering all major logger models, and for small and large clients including banks, healthcare companies, insurance carriers, law firms, and police departments.

XOVOX
114 Pearl St., Suite 2B
Port Chester, NY 10573 USA
+1-914-939-7396
www.xovox.tech

Note: this report is an updated version of a previous white paper entitled “Technical Description of Voice Logger Retrieval Techniques” issued in 2017 by Electrical Science, Inc.