What is the Standard Format for Digitized Audio?

Approaches for Storing Complex Audio Objects

Nick Krabbenhoeft
New York Public Library
United States of America
nickkrabbenhoeft@nypl.org

Abstract – The best practices for representing analog audio with digital bitstreams are relatively clear. Sample the signal with 24 bits of resolution at 96KHz. The standards for storing the data are less clear, especially for media with complex configurations of faces, regions, and streams. Whether accomplished through metadata and/or file format, the strategy chosen to represent the complexity of the original media has long-term preservation implications. Best practice guides rarely document these edge cases and informal discussions with practitioners have revealed a wide range of practices. This paper aims to outline the specific challenges of representing complex audio objects after digitization and approaches that have been implemented but not widely adopted.

Keywords – Audio, Digitization, Object Modeling
Conference Topics – Collaboration: a Necessity, an Opportunity or a Luxury?; Building Capacity, Capability and Community

I. Introduction

In response to the deteriorating sustainability of magnetic media, many organizations are pursuing digitization as their preservation strategy for audio and video collections. The New York Public Library has opted for this approach and has digitized over 200,000 objects in order to maintain the accessibility of their contents past the deterioration of the original media and/or playback equipment.

According to the OAIS Framework, organizations are responsible for defining the specifications for SIPs and AIPs, including the Content Objects those packages contain. Community guidelines such as IASA TC-04 simplify the process of defining Content Objects. For example, in order to represent the original audio signal at an equal or higher fidelity to what human ears can distinguish (20kHz), guidelines recommend sampling audio signals at a minimum of 44.1 kHz and at even higher rates to capture qualities of the recording medium outside the auditory range.

Best practice documents are less exact on how to store the bitstreams. Recommendations to keep audio signals as uncompressed PCM streams wrapped in a Wave or Broadcast Wave format leave room for interpretation. Some workflows store left and right stereo tracks in separate files while others may interleave them into a single file. Some workflows limit audio file sizes to 4 GB[1] while others use different file formats for long audio streams.

Reviews of the audio digitization literature have shown relatively little guidance on questions like this, and informal conversations have revealed a range of approaches. IASA TC-04 devotes three paragraphs in total to target formats. [1] In the Sound Directions project, Indiana University and Harvard University documented their approaches in greater detail, but they did differ. [2] As the scale of digitization increases, the number of situations not addressed within guidelines increase as well.

[1] The Wave file format based on the Resource Interchange File Format (RIFF), which allocates bytes 4-7 to specifying the file size. This limits the size to 2^32 bytes (about 4.295 GB). RF64 extension defined in EBU 3306 [7], allows for daisy chaining of additional audio data in 18 EB chunks.

16th International Conference on Digital Preservation
iPRES 2019, Amsterdam, The Netherlands.

This paper presents edge cases in digitized audio file specifications as encountered by the New York Public Library and documents potential options in hopes of spurring more public discussion.

II. The Challenge of Complex Audio Objects

Magnetic media is composed of metallic particles attached to a flexible tape by a binder. This composition does not inherently limit how information is recorded to the media. Audio may be stored as a stream of information in any location or orientation along this magnetic tape. The dependence on equipment for recording and playback equipment restricts the possibilities, but there is still great variety possible in the usage of a given format.

For example, the Compact Cassette format initially debuted in the 1960s as a format to record dictation. Early machines record a sequence of audio linearly within the upper or lower half of the tape. At the end of the tape, the cassette is flipped and audio is recorded to the other half. (Table 1.A)

Stereo content such as commercial music has a very similar layout, except the area used to record a single stream of mono audio is divided into narrower areas for left and right channels with a gap in between. (Table 1.B)

Other machines allow Compact Cassettes to be used as relatively low-cost studio recorders. Up to 4 inputs can be recorded simultaneously onto a tape, each perhaps representing an instrument like vocals, guitar, bass, and drums. The areas are arranged much like a 2-sided stereo cassette, except the tape is recorded in only one direction. (Table 1.C)

Finally, layouts can be a mixture of the above examples. Any machine that supports Compact Cassettes can record to them regardless of their prior use. For example, a tape first used in a dictation machine and then used to record music from the radio would have a mixture of mono and stereo arrangements. (Table 1.D)

Discovering and responding appropriately to the layout of audio is an important skill of audio engineers engaged in preservation. Each portion of the layout must be extracted with machinery appropriate to the layout of the recorded signal. Colloquially, a number of terms are used, such as streams, tracks, and channels. Frustratingly, these terms are imprecise in usage. What some may consider 2 mono tracks other may call 1 stereo channel. This paper uses the following terminology as defined in AES-57. [3]

1. Stream - a single linear sequence of audio signals
2. Region - a group of streams to be played back synchronously
3. Face - a group of regions to be played back sequentially

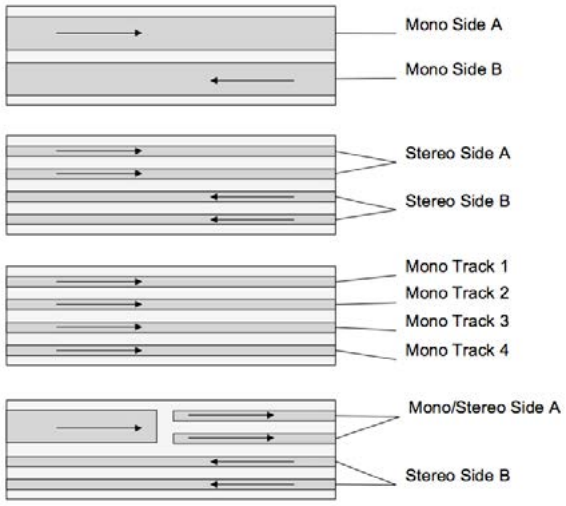

Table 1. Increasingly complex example layouts of audio on a Compact Cassette.
A dictation tape
A commercially released

Using those terms, the layouts in Table 1 would be described as follows:

A. 2 faces (Side A and B) each with 1 region and 1 stream
B. 2 faces each with 1 region and 2 streams
C. 1 face with 1 region and 4 streams
D. 2 faces. The first face has 2 regions. The first region has 1 stream and the second has 2 streams. The second face has 1 region and 2 streams
E. Strategies for Representing Complex Audio Objects

16th International Conference on Digital Preservation
iPRES 2019, Amsterdam, The Netherlands.

The complexity of Table 1.D demonstrates how a few choices made during recording can create intricate branching relationships. This is matched by the ability of collecting organizations to make choices on how to transform it into Content Information. Reviewed guidelines do not prescribe specific strategies. This section introduces potential strategies that use a combination of documentation and file formats.

To simplify this discussion, options will be illustrated through example objects with the following layouts:

- 1 face, 1 region, 1 stream
- 1 face, 1 region, multiple streams
- 1 face, multiple regions
- Multiple faces

1 face, 1 region, 1 stream

Example Item: oration on an open-reel audio tape recorded as mono

Even in this base example, the Wave and AIFF formats generally recommended for use may not be appropriate due to technical limitations.

The base specification for Wave and AIFF files stores the total file size as a 4-byte, unsigned integer in bytes 5-8 of the file. [4] [5] As a result, these formats are limited to a valid file size of 2^32 bytes (4 GiB or 4.295 GB) or 4 hours and 9 minutes of audio digitized at the typical digitization specifications of 96 kHz/24-bit. This length of mono is rare, but NYPL holds examples such as recordings of long-form speeches.

Assuming that a digitization program will produce valid files and retain a given audio, there are two potential strategies.

1. split the audio across multiple files
2. use a different file format

Creating multiple files per audio stream requires two additional considerations. First, the choices of how to divide the data such as appropriate point in the stream and whether or not to include overlap. These are not trivial choices and may require discussions with engineers on a case-by-case basis. Second, there are multiple methods to store the relationship between the files.

Splitting audio data across multiple files means the relationship between the object and the files is no longer obvious. To address this, a file naming convention can be modified to include the information, for example "part1", "part2", etc, but this should not be the sole form of metadata. As advised in Sound Directions, "filenames are not a reliable means of storing information." Filenames are directly editable from the file manager level as opposed to embedded metadata or metadata stored in sidecar files, and so they are more vulnerable to keying errors and accidental edits.

Major audiovisual metadata standards such as AES-57, PBCore, EBUCore, and AudioMD do not directly address situation, but generic structures within them that define one-to-many object-to-sub-object relationships could be applied. For example, the <pbcoreInstantiation> element can be used to describe any "unit that typically (though not always) comprises a whole representation of the asset." [6] Similarly, the <file> element in METS could be used to document the relationship between these files. [7] But, these methods are generally hypothetical and do not appear in literature. With digitization dating back to 2005, NYPL's metadata strategy is roughly based on AES-57. [8]

The second option for storing long durations of audio is to use another format. The Interchange File Format (IFF) that underlies AIFF and WAV was originally created in 1985. The technical constraints and assumptions of that era were fossilized into the specifications, file size being the most obvious of these.[1]

Extension specifications have been created for both AIFF and Wave that expand the total possible file size to 16EiB (roughly 2 million years of 96/24 audio). Sony published the Wave64 extension in 2003. Apple published the CAF extension for AIFF in 2005. and EBU published the RF64 extension in 2008. [9] [10] However, none of these extensions have received universal uptake in software used for audio digitization as an export option and many still consider them to be entirely different formats.

[1] New formats based on IFF continue to be developed and supported. For example, the WebP image format published by Google in 2011 has the same hard-coded 4GB size limit.

16th International Conference on Digital Preservation
iPRES 2019, Amsterdam, The Netherlands.

As an example, trying to export 4GB of audio from Audacity to Wave or AIFF prompts an error message. To create an RF64 or CAF, a user must find a separate export menu that lists those formats as options. Other programs default to ignoring the IFF file size metadata limitation and create invalid files in the base format specification.

Another possibility is to use another format entirely. More recent formats such as FLAC, MXF, and Matroska address the size issue in their base specification. [11] [12] 13] Again, there is uneven support for these formats within software used for audio digitization. However if they are supported, the file size issue is unambiguous audio unlike RF64, and they enable additional preservation friendly features such as embedded checksums and lossless compression. [14]

1 face, 1 region, multiple streams

Example Item: studio recording on an open-reel audio tape with 24 simultaneous tracks for different instruments

In audio file formats, streams are typically stored as channels corresponding to the expected speaker output. Wave and AIFF both natively support the most common multi-stream arrangement, stereo. But, they are poorly suited for storing more streams. Because the file size limit applies to the entire file, total possible duration decreases inversely to the number of streams. A WAV can hold roughly 4 hours of a single stream of 96/24 audio but only 1 hour of 4 simultaneous streams of 96/24 audio. Additionally, because larger number of streams are associated with surround sound speaker setups, the default file interpretation may not match the context of the original audio.

For example, in music studio production use, a stream or group of streams would capture a single instrument or voice during a recording session. This allows the instruments to be edited individually before being mixed down into a single stereo song. These streams are not intended to be played simultaneously without further mixing

Other formats support additional more complex layouts, including the MWBF extension to Wave, MXF, and Matroska.

The formats allow for further abstraction of audio arrangement through a concept called tracks. Multiple streams can be grouped as channels within a track separate from other streams while maintaining a synched timing. For example in a studio recording, instruments may be captured as a mixture of mono and stereo. Tracks can be used to keep organize this data within the file. As with any format choice, the biggest hurdle is ensuring export support from authoring software.

In production workflows, a common strategy is to save each stream to its own file. Used in a preservation workflow, this avoids format support issues, in exchange for requiring a metadata schema that records the relationships between files. It also requires specifying when to employ this strategy. Stereo audio is a multi-stream format. Interpreted stringently, a 1-file-per-stream strategy would save left and right audio streams were saved to separate files, instead of interleaving them.

1 face, multiple regions

Example Item: open-reel audio tape used to record sessions of dictation (mono) at different speeds

For media with regions, engineers must adjust the setup of playback equipment in accordance with the changing characteristics of the layout. Each of the changes, such as swapping a mono head for a 4-track head or adjusting the playback speed from 7.5 inches per second to 15 inches per second, requires stopping the playback process. Many workflows also require capturing the following audio as a new digital object.

Both Wave and AIFF support only a single audio chunk per file. If it is important to maintain a distinction between audio data from different regions, Wave and AIFF require creating a file per region. As with other multi-file strategies this also requires support in the metadata schema for maintaining the relationship between files. An interesting feature in Broadcast Wave is the TimeReference field that can be used to record the temporal relationship between two files on a shared timeline. [15

16th International Conference on Digital Preservation
iPRES 2019, Amsterdam, The Netherlands.

Container formats such as MXF and Matroska define an abstraction to demarcate playback often called a chapter. This provides the ability to sequence playback of the tracks within the container using chapter metadata. However, chapters assume sequential playback. During digitization, engineers will overlap the beginning and ends of neighboring regions to ensure total information capture. Experiments with container formats have not yielded a strategy for creating an unambiguous shared timeline within the container.

Multiple faces

Example Item: open-reel audio tape recorded in mono across four tracks

Faces have a sequential relationship, and the recommended strategy has been to store each face as a separate file. Although storing faces as chapters in a container file is a potential strategy, the difficulties in using chapters for regions would greatly complicate the representation of any audio with both faces and regions.

III. Discussion

There is a garden of forking paths when it comes to storing digitized audio. It would be helpful for digitization guidelines to go past 96/24 BWF recommendation and present options for file structures, but examples are difficult to find in the literature. Greater discussion and documentation of the approaches above would be particularly useful for two communities, digitization labs and repository developers.

In the first instance, the support for custom metadata formats, embedded metadata, Wave extensions, and container formats varies across digitization software and vendors. If every collecting institution chooses its own combination of strategies, labs are forced to support a wide range of strategies, increasing expense and likelihood of confusion or errors. After digitizing materials through in-house and vendor workflows, complex audio configurations is still a difficult class of media to design QC processes for. Documentation of even a few shared strategies would greatly simplify target selection for collecting organizations and support for labs.

In the second instance, representing the semantic relationship between files is one of the most challenging aspects of repository development. Documenting edge cases and migrating from previous strategies occupy outsized portions of time. Again, complex audio has presented a particular challenge for the development of ingest workflows at NYPL and, based on conversations, at other institutions as well.

While all of the summarized strategies are viable, it is from this perspective that the author finds container formats to be most worth investigation. NYPL has experimented with using the Matroska format to store 24 tracks of mono audio from a studio recording in a single file with an image of the track-listing. Doing so proved to be far simpler for object modeling than storing the relational metadata in a sidecar and developing a parser. However, as an experiment, it bears examination if such strategies impede access in the future.

This paper has discussed only strategies of how to reflect the structure of the physical object in a digital form. It does not discuss how intellectual content interacts with this organization. The layout of intellectual content might be entirely defined by the physical layout, such as the two sides of a tape being used to record different meetings. It may cut across the layout, such as a speech captured across two regions when it became the recording speed had to be lowered before the tape ran out. It may exist within the layout, like songs on a compact cassette. And it is most often a combination of the two. For the preservation of the original media, this paper advocates the primacy of the physical layout in creating digital objects while leaving the intellectual layout to presentation frameworks such as IIIF.

IV. Conclusion

This paper is a provocation to discuss and document how digitization projects encode and package outputs. It does not believe there is a single optimal strategy but hopes that as the scale audio digitization continues increasing and classes of edge start numbering in the thousands that common strategies may be developed.

16th International Conference on Digital Preservation
iPRES 2019, Amsterdam, The Netherlands.

References

[1] TC-04 Guidelines on the Production and Preservation of Digital Audio Objects (web edition). IASA, https://www.iasa-web.org/tc04/key-digital-principles.

[2] Sound Directions. Indiana University and Harvard University. http://www.dlib.indiana.edu/projects/sounddirections/.

[3] AES 57 AES standard for audio metadata - Audio object structures for preservation and restoration. Audio Engineering Society. http://www.aes.org/publications/standards/search.cfm?docID=84

[4] Wave 1.0 Specification. http://www-mmsp.ece.mcgill.ca/Documents/AudioFormats/WAVE/Docs/riffmci.pdf

[5] AIFF 1.3 Specification. http://www-mmsp.ece.mcgill.ca/Documents/AudioFormats/AIFF/Docs/AIFF-1.3.pdf

[6] PBCore Part Type, PBCore. https://pbcore.org/elements/pbcorepart

[7] METS. http://www.loc.gov/standards/mets/

[8] AMI Metadata Analog Reel Sample, NYPL. https://github.com/NYPL/ami-metadata/blob/master/versions/2.0/sample/sample_digitized_audioreelanalog.json#L48

[9] Multiple Channel Audio and Wave Files, Microsoft. http://www.microsoft.com/whdc/device/audio/multichaud.mspx

[10] EBU 3306 MBWF/RF64, European Broadcasting Union. https://tech.ebu.ch/publications/tech3306

[11] FLAC Specification. https://xiph.org/flac/format.html

[12] SMPTE MXF RDD 48. http://www.digitizationguidelines.gov/guidelines/rdd48-2018_published.pdf

[13] Matroska Specification. https://www.matroska.org/technical/specs/index.html

[14] Rice, Dave. Reconsidering the Checksum for Audiovisual Preservation. https://dericed.com/papers/reconsidering-the-checksum-for-audiovisual-preservation/

[15] A Primer on the Use of TimeReference, AVP. https://www.avpreserve.com/wp-content/uploads/2017/07/AVPS_TimeReference_Primer.pdf

16th International Conference on Digital Preservation
iPRES 2019, Amsterdam, The Netherlands.

iPRES 2019