

Internet Engineering Task Force  
Internet-Draft  
Intended status: Informational  
Expires: 6 October 2026

ssw. Whited, Ed.  
Independent  
4 April 2026

Matroska Stem Files  
draft-swhited-mka-stems-08

## Abstract

This document defines a multi-track profile of the Matroska container format for distributing stems. It is intended to be used by DJ applications and Digital Audio Workstations while remaining backwards compatible with existing media players.

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 6 October 2026.

## Copyright Notice

Copyright (c) 2026 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

## Table of Contents

|                                       |    |
|---------------------------------------|----|
| 1. Introduction . . . . .             | 2  |
| 1.1. Requirements Language . . . . .  | 3  |
| 2. Requirements . . . . .             | 3  |
| 3. Track Layout . . . . .             | 3  |
| 3.1. Audio Streams . . . . .          | 3  |
| 4. Digital Signal Processor . . . . . | 5  |
| 4.1. Compressor Metadata . . . . .    | 5  |
| 4.2. Limiter Metadata . . . . .       | 6  |
| 5. Format Support . . . . .           | 6  |
| 6. IANA Considerations . . . . .      | 7  |
| 7. Security Considerations . . . . .  | 8  |
| 8. Normative References . . . . .     | 8  |
| 9. Informative References . . . . .   | 9  |
| Acknowledgements . . . . .            | 9  |
| Author's Address . . . . .            | 10 |

## 1. Introduction

Stems are recordings of individual instruments, or clusters of instruments, used by DJs and music producers for live mixing of music. Historically stems have been stored as individual audio files, or using patent-encumbered or vendor specific, proprietary container formats.

A common feature of modern software used by DJs is "dynamic" or "live" stem separation where the DJ software attempts to algorithmically separate the audio signals in a track to allow the DJ to mute, solo, or apply effects to individual instruments. The results of such dynamic separation vary but are, generally speaking, noticeably different from the original stems used by the producer and frequently contain distortions and other artifacts that sound undesirable. A better model is to have the producer release the original stems and information about the mastering alongside the original track. This allows the final mix to sound closer to the producers original vision for the track, even while it is being remixed and interpreted by the DJ or another remixer.

This specification documents a profile for the Matroska container format [RFC9559] that allows it to store the final mix for a track alongside the lossless or lossy stems used to mix the track in a single file. In addition it specifies metadata for storing mastering information so that remixes using the stems can remain as close as possible to the producer of the tracks original intent. The target consumer of these stem files are DJ applications meant for live remixing and performance, as well as Digital Audio Workstations (DAWs) used by producers who want their music to be remixed.

### 1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

## 2. Requirements

STEM files have a few basic requirements including:

- \* Backwards compatibility with existing media players,
- \* The ability to store multiple audio track,
- \* The ability to store file-level metadata and track-level metadata, and
- \* Backwards compatibility when additional tracks have unknown formats that cannot be decoded.

## 3. Track Layout

### 3.1. Audio Streams

Each stem file may contain an arbitrary number of tracks containing audio and MUST include at least three audio tracks (the mixed audio and at least two stems). For stem files meant for live DJ use, it is RECOMMENDED that four or fewer stem tracks be used (as opposed to stem files meant for music production or non-live remixing where a DAW may utilize a significantly larger number of tracks).

For ease of decoding each track SHOULD be encoded using the same codec with the same parameters including bitrate, and sample rate. Stems are often recorded with a single channel and only the final mix is in stereo. Stems MAY have a different channel count or layout than the main audio track, however it is RECOMMENDED that all stem tracks maintain the same channel count and layout as the main track and have the same channel balance as their component parts in the final mix. For example, if the final mix is a stereo track that contains a fiddle that is 75% in the right channel and only 25% in the left channel, the stem track for the fiddle would also be in stereo with the stem mostly appearing from the right channel as in the final mix.

The first track containing audio data MUST be the final post-mix audio in the default language. All tracks containing the final post-mix audio regardless of language MUST have the Matroska "Default" flag set to "1" ([RFC9559], Section 18.1, 5.1.4.1.5). This helps preserve backwards compatibility in media players which do not support this format which typically play the first audio stream found or may select based on the default flag. In addition, the "Enabled" flag for any main tracks MUST be set to "1" ([RFC9559], Section 5.1.4.1.4).

The remaining audio tracks will be individual stems and MUST have the same effective length as the first track such that playing each stem track from the beginning would result in the same audio (excluding mastering) as the final mix present in the first track. For example, if the original track is three minutes long and the stem file includes a percussion track but the percussion does not start until minute two the percussion stem would still be three minutes long but would contain a minute of silence at the start of the track, or would have a block timestamp ([RFC9559], Section 10) that sets the effective start time to one minute.

Each stem track MUST have the Matroska "Default" flag set to "0" and MUST have the "Enabled" flag set to "0".

The stem tracks SHOULD NOT have any gain normalization applied to bring the stems up to the same perceived volume. Instead they should retain the same levels as they would have in the final mix present in the default track so that if all stems were played at unity gain the overall level would be equivalent to the level of the final mix.

Each stem track (ie. all tracks that are not the first track) MUST set the value of the track Name element ([RFC9559], Section 5.1.4.1.18) to a short, human-meaningful, track name for the stem that describes its contents, for example "Percussion" or "Vocals". These names are intended for display in playback applications and therefore should remain concise (generally no more than one word), but no specific format or length requirement is defined. The track Name element MAY also be duplicated or overridden as a tag, in which case the order of precedence from Section 24.1 of [RFC9559] SHOULD be respected.

For each stem track a tag ([RFC9559], Section 5.1.8) SHOULD also be set with its target set to the stem track and a tag name of "STEM\_COLOR". The tag value must be a string in RGB hex format set to a color representing the stem (ie. #145374).

#### 4. Digital Signal Processor

Because mastering happens post-mix and the stems are pre-mix audio the stem tracks SHOULD NOT have any mastering steps applied. Instead, metadata for configuring a compressor and limiter SHOULD be included in the file's global metadata as simple tags (see Section 5.1.8.1.2 of [RFC9559]). After mixing, playback applications MAY choose to feed the mix through a Digital Signal Processor (DSP) configured with the limiter and compressor settings read from the metadata.

Each binary setting for the compressor or limiter is stored as a floating-point number in the 32-bit and 64-bit binary interchange format, as defined in [IEEE\_754\_2019] with the additional restriction that they are limited to a minimum value of 0.0 and a maximum value of 1.0. Because different DSPs may use different ranges or scales for each value the playback software SHOULD interpret the 0-1 values as a linear scale and map them to the range and scale required by the DSP when configuring the DSP for playback. This may result in a loss of fidelity on some DSPs, but this is deemed an acceptable trade off for stem playback which would not normally be able to have a mastering step at all.

During production of a stem track, vendor specific metadata MAY be embedded in the Matroska file for more accurately configuring a specific DSP, but if such metadata is included the scaled values SHOULD also be present for those without access to the specific DSP used for the track and such metadata MUST select tag names in such a way that they do not conflict with the tag names defined for the generic compressor or limiter.

##### 4.1. Compressor Metadata

| Tag Name               | Type   | Values            |
|------------------------|--------|-------------------|
| COMPRESSOR_ENABLED     | UTF-8  | "TRUE" or "FALSE" |
| COMPRESSOR_RATIO       | binary | 0.0-1.0           |
| COMPRESSOR_OUTPUT_GAIN | binary | 0.0-1.0           |
| COMPRESSOR_THRESHOLD   | binary | 0.0-1.0           |
| COMPRESSOR_ATTACK      | binary | 0.0-1.0           |
| COMPRESSOR_INPUT_GAIN  | binary | 0.0-1.0           |

|                       |         |         |         |
|-----------------------|---------|---------|---------|
| COMPRESSOR_RELEASE    | binary  | 0.0-1.0 |         |
| +-----+               | +-----+ | +-----+ | +-----+ |
| COMPRESSOR_HP_CUTOFF  | binary  | 0.0-1.0 |         |
| +-----+               | +-----+ | +-----+ | +-----+ |
| COMPRESSOR_HP_DRY_WET | binary  | 0.0-1.0 |         |
| +-----+               | +-----+ | +-----+ | +-----+ |

Table 1: Compressor metadata tags

#### 4.2. Limiter Metadata

|                   |         |                   |         |
|-------------------|---------|-------------------|---------|
| +=====+           | +=====+ | +=====+           | +=====+ |
| Tag Name          | Type    | Values            |         |
| +-----+           | +-----+ | +-----+           | +-----+ |
| LIMITER_ENABLED   | UTF-8   | "TRUE" or "FALSE" |         |
| +-----+           | +-----+ | +-----+           | +-----+ |
| LIMITER_RELEASE   | binary  | 0.0-1.0           |         |
| +-----+           | +-----+ | +-----+           | +-----+ |
| LIMITER_THRESHOLD | binary  | 0.0-1.0           |         |
| +-----+           | +-----+ | +-----+           | +-----+ |
| LIMITER_CEILING   | binary  | 0.0-1.0           |         |
| +-----+           | +-----+ | +-----+           | +-----+ |

Table 2: Limiter metadata tags

#### 5. Format Support

The Matroska container format can store many types of audio, not all of which are suitable for DJing or music production. To ensure compatibility between playback and encoding applications the following formats SHOULD be supported depending on the use case of the software as shown in the following table. Formats with the use case "Live remixing" are intended largely for playback applications meant for live performance (ie. DJ software). Formats with the use case "Music production" are intended to be distributed for remixing in a non-live setting (ie. with a DAW or music tracker).

| Codec                                     | Use Case                           | Codec ID   |
|---|------------------------------------|--|
| FLAC [RFC9639]                            | Live remixing,<br>Music production | A_FLAC [RFC9639],<br>Section 10.2                              |
| Opus [RFC6716]                            | Live remixing                      | A_OPUS<br>[I-D.ietf-cellar-codec],<br>Section 3.4.32           |
| Raw PCM (IEEE<br>float, little<br>endian) | Music production                   | A_PCM/FLOAT/IEEE<br>[I-D.ietf-cellar-codec],<br>Section 3.4.33 |
| Raw PCM<br>(integer, big<br>endian)       | Music production                   | A_PCM/INT/BIG<br>[I-D.ietf-cellar-codec],<br>Section 3.4.34    |
| Raw PCM<br>(integer,<br>little endian)    | Music production                   | A_PCM/INT/LIT<br>[I-D.ietf-cellar-codec],<br>Section 3.4.35    |

Table 3: Audio codec support

## 6. IANA Considerations

This memo modifies the "Matroska Tag Names" registry to add the following values:

| Tag Name               | Tag Type | Reference                  |
|------------------------|----------|----------------------------|
| STEM_COLOR             | UTF-8    | This document, Section 3.1 |
| COMPRESSOR_ENABLED     | UTF-8    | This document, Section 4.1 |
| COMPRESSOR_RATIO       | binary   | This document, Section 4.1 |
| COMPRESSOR_OUTPUT_GAIN | binary   | This document, Section 4.1 |
| COMPRESSOR_THRESHOLD   | binary   | This document, Section 4.1 |
| COMPRESSOR_ATTACK      | binary   | This document, Section 4.1 |
| COMPRESSOR_INPUT_GAIN  | binary   | This document, Section 4.1 |
| COMPRESSOR_RELEASE     | binary   | This document, Section 4.1 |
| COMPRESSOR_HP_CUTOFF   | binary   | This document, Section 4.1 |
| COMPRESSOR_HP_DRY_WET  | binary   | This document, Section 4.1 |
| LIMITER_ENABLED        | UTF-8    | This document, Section 4.2 |
| LIMITER_RELEASE        | binary   | This document, Section 4.2 |
| LIMITER_THRESHOLD      | binary   | This document, Section 4.2 |
| LIMITER_CEILING        | binary   | This document, Section 4.2 |

Table 4: Additions to the "Matroska Tag Names" Registry

## 7. Security Considerations

This document inherits security considerations from both [RFC8794] and [RFC9559]. It does not have additional security considerations.

## 8. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.

- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC9559] Lhomme, S., Bunkus, M., and D. Rice, "Matroska Media Container Format Specification", RFC 9559, DOI 10.17487/RFC9559, October 2024, <<https://www.rfc-editor.org/info/rfc9559>>.

## 9. Informative References

- [RFC6716] Valin, JM., Vos, K., and T. Terriberry, "Definition of the Opus Audio Codec", RFC 6716, DOI 10.17487/RFC6716, September 2012, <<https://www.rfc-editor.org/info/rfc6716>>.
- [IEEE\_754\_2019] IEEE, "IEEE Standard for Floating-Point Arithmetic", IEEE IEEE 754-2019, DOI 10.1109/IEEESTD.2019.8766229, 18 July 2019, <<https://ieeexplore.ieee.org/document/8766229>>.
- [RFC8794] Lhomme, S., Rice, D., and M. Bunkus, "Extensible Binary Meta Language", RFC 8794, DOI 10.17487/RFC8794, July 2020, <<https://www.rfc-editor.org/info/rfc8794>>.
- [RFC9639] van Beurden, M.Q.C. and A. Weaver, "Free Lossless Audio Codec (FLAC)", RFC 9639, DOI 10.17487/RFC9639, December 2024, <<https://www.rfc-editor.org/info/rfc9639>>.
- [I-D.ietf-cellar-codec] Lhomme, S., Bunkus, M., and D. Rice, "Matroska Media Container Codec Specifications", Work in Progress, Internet-Draft, draft-ietf-cellar-codec-17, 15 February 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-cellar-codec-17>>.

## Acknowledgements

Thanks to the members of #matroska on the libera.chat IRC network, and to mosu and JanC in particular, for patiently explaining the basics of the format to me and for all their feedback.

Thanks also to the members of the Ardour forums for their feedback on DAWs and mastering.

Finally, thanks to the members of the IETF CELLAR working group, especially Steve Lhomme, for their feedback.

Author's Address

Sam Whited (editor)  
Independent  
Email: [sam@samwhited.com](mailto:sam@samwhited.com)  
URI: <https://blog.samwhited.com>