
audio-metadata

Release 0.11.1

thebigmunch

May 14, 2020

CONTENTS

1	Getting Started	3
2	Overview	5
2.1	API Reference	6
	Index	15

audio-metadata is a library for reading and, in the future, writing audio metadata.

GETTING STARTED

Install audio-metadata with `pip`.

```
$ pip install -U audio-metadata
```


OVERVIEW

The goals of audio-metadata are to provide a nice API and good UX while keeping the codebase as clean and simple as possible.

Features and functionality that set it apart:

- **Uses the Python standard load(s)/dump(s) API.**
 - Can load filepaths, os.PathLike objects, file-like objects, and bytes-like objects.
- **Metadata objects look like a dict and act like a dict.**
 - Some common libraries shadow the representation of a dict and/or dict methods but do not behave like a dict.
 - Supports attribute-style access that can be mixed with dict key-subscription.
- **All metadata objects have user-friendly representations.**
 - This includes *humanized* representations of certain values like filesize, bitrate, duration, and sample rate.

```
>>> import audio_metadata

>>> metadata = audio_metadata.load('05 - Heart of Hearts.flac')

>>> metadata
<FLAC ({
  'filepath': '05 - Heart of Hearts.flac',
  'filesize': '44.23 MiB',
  'pictures': [],
  'seektable': <FLACSeekTable (37 seekpoints)>,
  'streaminfo': <FLACStreamInfo ({
    'bit_depth': 16,
    'bitrate': '1022 Kbps',
    'channels': 2,
    'duration': '06:03',
    'md5': '3ae700893d099a5d281a5d8db7847671',
    'sample_rate': '44.1 KHz',
  })>,
  'tags': <VorbisComment ({
    'album': ['Myth Takes'],
    'artist': ['!!!!'],
    'bpm': ['119'],
    'date': ['2007'],
    'genre': ['Dance Punk'],
    'title': ['Heart of Hearts'],
    'tracknumber': ['05'],
```

(continues on next page)

(continued from previous page)

```

    })>,
  })>

>>> metadata['streaminfo']
<FLACStreamInfo ({
  'bit_depth': 16,
  'bitrate': '1022 Kbps',
  'channels': 2,
  'duration': '06:03',
  'md5': '3ae700893d099a5d281a5d8db7847671',
  'sample_rate': '44.1 KHz',
})>

>>> metadata.streaminfo.bitrate
1022134.0362995076

>>> metadata.streaminfo['duration']
362.90666666666667

>>> metadata['streaminfo'].sample_rate
44100

```

See the full [API Reference](#).

2.1 API Reference

The main methods of interacting with audio metadata are *load* and *loads*.

2.1.1 Core

`audio_metadata.determine_format(data)`

Determine the format of a filepath, file-like object, or bytes-like object.

Parameters *data* (*bytes-like object*, *str*, *os.PathLike*, or *file-like object*) – A bytes-like object, filepath, path-like object or file-like object of an audio file.

Returns An appropriate audio format class if supported, else None.

Return type *Format*

`audio_metadata.load(f)`

Load audio metadata from a filepath or file-like object.

Parameters *f* (*str*, *os.PathLike*, or *file-like object*) – A filepath, path-like object or file-like object of an audio file.

Returns An audio format object of the appropriate type.

Return type *Format*

Raises

- *FormatError* – If the audio file is not valid.
- *UnsupportedFormat* – If the audio file is not of a supported format.
- *ValueError* – If *f* is not a valid str, path-like object, file-like object, or is unreadable.

`audio_metadata.loads(b)`

Load audio metadata from a bytes-like object.

Parameters `b` (*bytes-like object*) – A bytes-like object of an audio file.

Returns An audio format object of the appropriate type.

Return type *Format*

Raises

- *FormatError* – If the audio file is not valid.
- *UnsupportedFormat* – If the audio file is not of a supported format.
- *ValueError* – If `b` is not a valid bytes-like object.

2.1.2 Exceptions

exception `audio_metadata.AudioMetadataException`

Base exception for audio-metadata.

exception `audio_metadata.FormatError`

The binary format of a data input is invalid.

exception `audio_metadata.TagError`

A tag is not compliant to a specification.

exception `audio_metadata.UnsupportedFormat`

An unsupported format, version, or profile was encountered.

2.1.3 Base Classes

class `audio_metadata.Format`

Base class for audio format objects.

filepath

Path to audio file, if applicable.

Type *str*

filesize

Size of audio file.

Type *int*

pictures

A list of *Picture* objects.

Type *list*

tags

A *Tags* object.

Type *Tags*

class `audio_metadata.Picture(mapping=None, **kwargs)`

Base class for picture objects.

class `audio_metadata.StreamInfo(mapping=None, **kwargs)`

Base class for stream information objects.

class `audio_metadata.Tags` (*mapping=None, **kwargs*)

Base class for tags objects.

FIELD_MAP

A mapping of format-specific field names to common aliases.

Type `frozenbidict`

2.1.4 FLAC

class `audio_metadata.FLAC`

FLAC file format object.

Extends *Format*.

cuesheet

The cuesheet metadata block.

Type *FLACCueSheet*

pictures

A list of *FLACPicture* objects.

Type `list`

seektable

The seektable metadata block.

Type *FLACSeekTable*

streaminfo

The audio stream information.

Type *FLACStreamInfo*

tags

The Vorbis comment metadata block.

Type *VorbisComments*

class `audio_metadata.FLACApplication` (**, id, data*)

A FLAC application metadata block.

id

The 32-bit application identifier.

Type `str`

data

The data defined by the application.

Type `bytes`

class `audio_metadata.FLACCueSheet` (*tracks, catalog_number, lead_in_samples, compact_disc*)

A FLAC cue sheet metadata block.

A list-like structure of *FLACCueSheetTrack* objects along with some information used in the cue sheet.

catalog_number

The media catalog number.

Type `str`

lead_in_samples

The number of lead-in samples. This is only meaningful for CD-DA cuesheets. For others, it should be 0.

Type `int`

compact_disc

True if the cue sheet corresponds to a compact disc, else False.

Type `bool`

class `audio_metadata.FLACCueSheetIndex` (*, *number*, *offset*)

A FLAC cue sheet track index point.

number

The index point number.

The first index in a track must have a number of 0 or 1.

Index numbers must increase by 1 and be unique within a track.

For CD-DA, an index number of 0 corresponds to the track pre-gab.

Type `int`

offset

Offset in samples relative to the track offset.

Type `int`

class `audio_metadata.FLACCueSheetTrack` (*, *track_number*, *offset*, *isrc*, *type*, *pre_emphasis*, *indexes=NOTHING*)

A FLAC cue sheet track.

track_number

The track number of the track.

0 is not allowed to avoid conflicting with the CD-DA spec lead-in track.

For CD-DA, the track number must be 1-99 or 170 for the lead-out track.

For non-CD-DA, the track number must be 255 for the lead-out track.

Track numbers must be unique within a cue sheet.

Type `int`

offset

Offset in samples relative to the beginning of the FLAC audio stream.

Type `int`

isrc

The ISRC (International Standard Recording Code) of the track.

Type `str`

type

0 for audio, 1 for non-audio.

Type `int`

pre_emphasis

True if contains pre-emphasis, False if not.

Type `bool`

indexes

The index points for the track as *FLACCueSheetIndex* objects.

Type `list`

```
class audio_metadata.FLACMetadataBlock(* , type, data)
    Generic FLAC metadata block.

    type
        Metadata block type index.
        Type int

    data
        The binary metadata block data.
        Type bytes

class audio_metadata.FLACPadding(* , size)
    A FLAC padding metadata block.

    size
        The size of the padding.
        Type int

class audio_metadata.FLACPicture(mapping=None, **kwargs)
    A FLAC picture object.

    type
        The picture type according to the ID3v2 APIC frame format.
        Type ID3PictureType

    mime_type
        The mime type of the picture. May indicate that the picture data is an URL of the picture instead of picture
        data.
        Type str

    description
        The description of the picture.
        Type str

    width
        The width of the picture in pixels.
        Type int

    height
        The height of the picture in pixels.
        Type int

    bit_depth
        The color depth of the picture in bits-per-pixel.
        Type int

    colors
        For indexed-color pictures (e.g. GIF), the number of colors used. Should be 0 for non-indexed-color
        pictures.
        Type int

    data
        The binary picture data.
        Type bytes
```

```

class audio_metadata.FLACSeekPoint (*, first_sample, offset, num_samples)
class audio_metadata.FLACSeekTable (initlist=None)
class audio_metadata.FLACStreamInfo (*, start, size, min_block_size, max_block_size,
                                     min_frame_size, max_frame_size, bit_depth, bitrate,
                                     channels, duration, md5, sample_rate)
class audio_metadata.FLACVorbisComments (*args, **kwargs)

```

2.1.5 ID3v1

```

class audio_metadata.ID3v1 (mapping=None, **kwargs)
class audio_metadata.ID3v1Fields (mapping=None, **kwargs)

```

2.1.6 ID3v2

```

class audio_metadata.ID3v2 (mapping=None, **kwargs)
class audio_metadata.ID3v2Flags (*, unsync=False, compressed=False, extended=False, experi-
                                mental=False, footer=False)
class audio_metadata.ID3v2Frames (mapping=None, *, id3_version=<ID3Version.v24>,
                                **kwargs)
class audio_metadata.ID3v2Header (*, size, version, flags)

```

2.1.7 MP3

```

class audio_metadata.MP3
    MP3 file format object.

    Extends Format.

    pictures
        A list of ID3v2Picture objects.

        Type list

    streaminfo
        The audio stream information.

        Type MP3StreamInfo

    tags
        The ID3v2 tag frames, if present.

        Type ID3v2Frames

class audio_metadata.LAMEEncodingFlags (*, nogap_continuation, nogap_continued, nssafe-
                                       joint, nspsytune)

class audio_metadata.LAMEHeader (*, crc, version, revision, ath_type, audio_crc, audio_size,
                                bitrate, bitrate_mode, channel_mode, delay, encoding_flags,
                                lowpass_filter, mp3_gain, noise_shaping, padding, pre-
                                set, replay_gain, source_sample_rate, surround_info, un-
                                wise_settings_used)

class audio_metadata.LAMEReplayGain (*, peak, track_type, track_origin, track_adjustment, al-
                                    bum_type, album_origin, album_adjustment)

```

```
class audio_metadata.MP3StreamInfo(*, start, end, size, vbri, xing, version, layer, protected,
                                   bitrate, bitrate_mode, channel_mode, channels, duration,
                                   sample_rate)

class audio_metadata.MPEGFrameHeader(*, start, size, vbri, xing, version, layer, pro-
                                     tected, padded, bitrate, channel_mode, channels, sam-
                                     ple_rate)

class audio_metadata.VBRIHeader(*, version, delay, quality, num_bytes, num_frames,
                                num_toc_entries, toc_scale_factor, toc_entry_num_bytes,
                                toc_entry_num_frames, toc)

class audio_metadata.VBRIToC(initlist=None)

class audio_metadata.XingHeader(*, lame, num_frames, num_bytes, toc, quality)

class audio_metadata.XingToC(initlist=None)
```

2.1.8 Ogg

```
class audio_metadata.Ogg
    Ogg file format object.

    Extends Format.

    Base class for various formats using an Ogg container.

class audio_metadata.OggPage(*, header, is_complete, is_continued, is_first, is_last, position, se-
                              rial_number, sequence_number, crc, num_segments, segments)

class audio_metadata.OggPageHeader(*, start, version, is_continued, is_first, is_last, position, se-
                                   rial_number, sequence_number, crc, num_segments)

class audio_metadata.OggPageSegments(initlist=None)
```

2.1.9 Ogg Opus

```
class audio_metadata.OggOpus
    Ogg Opus file format object.

    Extends Format.

    pictures
        A list of FLACPicture objects.

        Type list

    streaminfo
        The audio stream information.

        Type OggOpusStreamInfo

    tags
        The Vorbis comment metadata block.

        Type OggOpusVorbisComments

class audio_metadata.OggOpusStreamInfo(*, start, size, version, bitrate, channel_map,
                                         channels, duration, output_gain, pre_skip, sam-
                                         ple_rate=48000, source_sample_rate)

class audio_metadata.OggOpusVorbisComments(*args, **kwargs)
```


2.1.10 Ogg Opus

class `audio_metadata.OggVorbis`
Ogg Vorbis file format object.

Extends *Format*.

pictures
A list of *FLACPicture* objects.

Type *list*

streaminfo
The audio stream information.

Type *OggVorbisStreamInfo*

tags
The Vorbis comment metadata block.

Type *OggVorbisComments*

class `audio_metadata.OggVorbisStreamInfo` (*, *start*, *size*, *version*, *bitrate*, *channels*, *duration*,
max_bitrate, *min_bitrate*, *nominal_bitrate*, *sample_rate*)

class `audio_metadata.OggVorbisComments` (*args, **kwargs)

2.1.11 Vorbis

class `audio_metadata.VorbisComment` (*, *value*, *name*)

class `audio_metadata.VorbisComments` (*args, **kwargs)

2.1.12 WAV

class `audio_metadata.WAVE`
WAVE file format object.

Extends *Format*.

pictures
A list of *ID3v2Picture* objects.

Type *list*

streaminfo
The audio stream information.

Type *WAVStreamInfo*

tags
The ID3v2 or RIFF tags, if present.

Type *ID3v2Frames* or *RIFFTags*

class `audio_metadata.RIFFTag` (*, *name*, *value*)

class `audio_metadata.RIFFTags` (*mapping=None*, **kwargs)

class `audio_metadata.WAVEStreamInfo` (*, *start*, *size*, *extension_data*, *audio_format*, *bit_depth*,
bitrate, *channels*, *duration*, *sample_rate*)

```
class audio_metadata.WAVESubchunk (*, id, data)
```

A

AudioMetadataException, 7

B

bit_depth (*audio_metadata.FLACPicture attribute*), 10

C

catalog_number (*audio_metadata.FLACCueSheet attribute*), 8

colors (*audio_metadata.FLACPicture attribute*), 10

compact_disc (*audio_metadata.FLACCueSheet attribute*), 9

cuesheet (*audio_metadata.FLAC attribute*), 8

D

data (*audio_metadata.FLACApplication attribute*), 8

data (*audio_metadata.FLACMetadataBlock attribute*), 10

data (*audio_metadata.FLACPicture attribute*), 10

description (*audio_metadata.FLACPicture attribute*), 10

determine_format() (in module *audio_metadata*), 6

F

FIELD_MAP (*audio_metadata.Tags attribute*), 8

filepath (*audio_metadata.Format attribute*), 7

filesize (*audio_metadata.Format attribute*), 7

FLAC (class in *audio_metadata*), 8

FLACApplication (class in *audio_metadata*), 8

FLACCueSheet (class in *audio_metadata*), 8

FLACCueSheetIndex (class in *audio_metadata*), 9

FLACCueSheetTrack (class in *audio_metadata*), 9

FLACMetadataBlock (class in *audio_metadata*), 9

FLACPadding (class in *audio_metadata*), 10

FLACPicture (class in *audio_metadata*), 10

FLACSeekPoint (class in *audio_metadata*), 10

FLACSeekTable (class in *audio_metadata*), 11

FLACStreamInfo (class in *audio_metadata*), 11

FLACVorbisComments (class in *audio_metadata*), 11

Format (class in *audio_metadata*), 7

FormatError, 7

H

height (*audio_metadata.FLACPicture attribute*), 10

I

id (*audio_metadata.FLACApplication attribute*), 8

ID3v1 (class in *audio_metadata*), 11

ID3v1Fields (class in *audio_metadata*), 11

ID3v2 (class in *audio_metadata*), 11

ID3v2Flags (class in *audio_metadata*), 11

ID3v2Frames (class in *audio_metadata*), 11

ID3v2Header (class in *audio_metadata*), 11

indexes (*audio_metadata.FLACCueSheetTrack attribute*), 9

isrc (*audio_metadata.FLACCueSheetTrack attribute*), 9

L

LAMEEncodingFlags (class in *audio_metadata*), 11

LAMEHeader (class in *audio_metadata*), 11

LAMEReplayGain (class in *audio_metadata*), 11

lead_in_samples (*audio_metadata.FLACCueSheet attribute*), 8

load() (in module *audio_metadata*), 6

loads() (in module *audio_metadata*), 6

M

mime_type (*audio_metadata.FLACPicture attribute*), 10

MP3 (class in *audio_metadata*), 11

MP3StreamInfo (class in *audio_metadata*), 11

MPEGFrameHeader (class in *audio_metadata*), 12

N

number (*audio_metadata.FLACCueSheetIndex attribute*), 9

O

offset (*audio_metadata.FLACCueSheetIndex attribute*), 9

- offset (*audio_metadata.FLACCueSheetTrack attribute*), 9
- Ogg (*class in audio_metadata*), 12
- OggOpus (*class in audio_metadata*), 12
- OggOpusStreamInfo (*class in audio_metadata*), 12
- OggOpusVorbisComments (*class in audio_metadata*), 12
- OggPage (*class in audio_metadata*), 12
- OggPageHeader (*class in audio_metadata*), 12
- OggPageSegments (*class in audio_metadata*), 12
- OggVorbis (*class in audio_metadata*), 13
- OggVorbisComments (*class in audio_metadata*), 13
- OggVorbisStreamInfo (*class in audio_metadata*), 13
- ## P
- Picture (*class in audio_metadata*), 7
- pictures (*audio_metadata.FLAC attribute*), 8
- pictures (*audio_metadata.Format attribute*), 7
- pictures (*audio_metadata.MP3 attribute*), 11
- pictures (*audio_metadata.OggOpus attribute*), 12
- pictures (*audio_metadata.OggVorbis attribute*), 13
- pictures (*audio_metadata.WAVE attribute*), 13
- pre_emphasis (*audio_metadata.FLACCueSheetTrack attribute*), 9
- ## R
- RIFFTag (*class in audio_metadata*), 13
- RIFFTags (*class in audio_metadata*), 13
- ## S
- seektable (*audio_metadata.FLAC attribute*), 8
- size (*audio_metadata.FLACPadding attribute*), 10
- streaminfo (*audio_metadata.FLAC attribute*), 8
- streaminfo (*audio_metadata.MP3 attribute*), 11
- streaminfo (*audio_metadata.OggOpus attribute*), 12
- streaminfo (*audio_metadata.OggVorbis attribute*), 13
- streaminfo (*audio_metadata.WAVE attribute*), 13
- StreamInfo (*class in audio_metadata*), 7
- ## T
- TagError, 7
- tags (*audio_metadata.FLAC attribute*), 8
- tags (*audio_metadata.Format attribute*), 7
- tags (*audio_metadata.MP3 attribute*), 11
- tags (*audio_metadata.OggOpus attribute*), 12
- tags (*audio_metadata.OggVorbis attribute*), 13
- tags (*audio_metadata.WAVE attribute*), 13
- Tags (*class in audio_metadata*), 7
- track_number (*audio_metadata.FLACCueSheetTrack attribute*), 9
- type (*audio_metadata.FLACCueSheetTrack attribute*), 9
- type (*audio_metadata.FLACMetadataBlock attribute*), 10
- type (*audio_metadata.FLACPicture attribute*), 10
- ## U
- UnsupportedFormat, 7
- ## V
- VBRIHeader (*class in audio_metadata*), 12
- VBRIToC (*class in audio_metadata*), 12
- VorbisComment (*class in audio_metadata*), 13
- VorbisComments (*class in audio_metadata*), 13
- ## W
- WAVE (*class in audio_metadata*), 13
- WAVEStreamInfo (*class in audio_metadata*), 13
- WAVESubchunk (*class in audio_metadata*), 13
- width (*audio_metadata.FLACPicture attribute*), 10
- ## X
- XingHeader (*class in audio_metadata*), 12
- XingToC (*class in audio_metadata*), 12