mkvextract -- extract tracks from *Matroska*(tm) files into other files

Table of contents

1. Synopsis

```
mkvextract {source-filename} {mode1} [options] [extraction-spec1] [mode2] [options] [extraction-spec2] […]
```

2. Description

This program extracts specific parts from a *Matroska*(tm) file to other useful formats. The first argument is the name of the source file which must be a *Matroska*(tm) file.

All other arguments either switch to a certain extraction mode, change options for the currently active mode or specify what to extract into which file. Multiple modes can be used in the same invocation of mkvextract allowing the extraction of multiple things in a single pass. Most options can only be used in certain modes with a few options applying to all modes.

Currently supported is the extraction of tracks, tags, attachments, chapters, CUE sheets, timestamps and cues.

2.1. Common options

The following options are available in all modes and only described once in this section.

Option	Description
`-f`, `--parse-fully`	Sets the parse mode to 'full'. The default mode does not parse the whole file but uses the meta seek elements for locating the required elements of a source file. In 99% of all cases this is enough. But for files that do not contain meta seek elements or which are damaged the user might have to use this mode. A full scan of a file can take a couple of minutes while a fast scan only takes seconds.
`--command-line-charset` *character-set*	Sets the character set to convert strings given on the command line from. It defaults to the character set given by system's current locale.
`--output-charset` *character-set*	Sets the character set to which strings are converted that are to be output. It defaults to the character set given by system's current locale.
`-r`, `--redirect-output` *file-name*	Writes all messages to the file *file-name* instead of to the console. While this can be done easily with output redirection there are cases in which this option is needed: when the terminal reinterprets the output before writing it to a file. The character set set with --output-charset is honored.
`--flush-on-close`	Tells the program to flush all data cached in memory to storage when closing files opened for writing. This can be used to prevent data loss on power outages or to circumvent certain problems in the operating system or drivers. The downside is that multiplexing will take longer as mkvmerge will wait until all data has been written to the storage before exiting. See issues #2469 and #2480 on the MKVToolNix bug tracker for in-depth discussions on the pros and cons.
`--ui-language` *code*	Forces the translations for the language *code* to be used (e.g. '`de_DE`' for the German translations). Entering '`list`' as the *code* will cause the program to output a list of available translations.
`--abort-on-warnings`	Tells the program to abort after the first warning is emitted. The program's exit code will be 1.
`--debug` *topic*	Turn on debugging for a specific feature. This option is only useful for developers.
`--engage` *feature*	Turn on experimental features. A list of available features can be requested with `mkvextract --engage list`. These features are not meant to be used in normal situations.
`--gui-mode`	Turns on GUI mode. In this mode specially-formatted lines may be output that can tell a controlling GUI what's happening. These messages follow the format '`#GUI#message`'. The message may be followed by key/value pairs as in '`#GUI#message#key1=value1#key2=value2…`'. Neither the messages nor the keys are ever translated and always output in English.
`-v`, `--verbose`	Be verbose and show all the important *Matroska*(tm) elements as they're read.
`-h`, `--help`	Show usage information and exit.
`-V`, `--version`	Show version information and exit.
`@`*options-file.json*	Reads additional command line arguments from the file *options-file*. For a full explanation on the supported formats for such files see the section called "Option files" in the *mkvmerge*(1) man page.

2.2. Track extraction mode

Syntax: `mkvextract source-filename tracks [options] TID1:dest-filename1 [TID2:dest-filename2 ...]`

The following command line options are available for each track in the '`tracks`' extraction mode. They have to appear in front of the track specification (see below) they should be applied to.

Option	Description
`-c character-set`	Sets the character set to convert the next text subtitle track to. Only valid if the next track ID targets a text subtitle track. It defaults to UTF-8.
`--blockadd Level`	Keep only the BlockAdditions up to this level. The default is to keep all levels. This option only affects certain kinds of codecs like WAVPACK4.
`--cuesheet`	Causes *mkvextract*(1) to extract a CUE sheet from the chapter information and tag data for the following track into a file whose name is the track's output name with '`.cue`' appended to it.
`--raw`	Extracts the raw data into a file without any container data around it. Unlike the --fullraw flag this flag does not cause the contents of the `CodecPrivate` element to be written to the file. This mode works with all `CodecIDs`, even the ones that *mkvextract*(1) doesn't support otherwise, but the resulting files might not be usable.
`--fullraw`	Extracts the raw data into a file without any container data around it. The contents of the `CodecPrivate` element will be written to the file first if the track contains such a header element. This mode works with all `CodecIDs`, even the ones that *mkvextract*(1) doesn't support otherwise, but the resulting files might not be usable.
`TID:outname`	Causes extraction of the track with the ID `TID` into the file `outname` if such a track exists in the source file. This option can be given multiple times. The track IDs are the same as the ones output by *mkvmerge*(1)'s `--identify` option. Each output name should be used only once. The exception are RealAudio and RealVideo tracks. If you use the same name for different tracks then those tracks will be saved in the same file. Example: `$ mkvextract input.mkv tracks 0:video.h264 2:output-two-vobsub-tracks.idx 3:output-two-vobsub-tracks.idx`

2.3. Attachments extraction mode

Syntax: `mkvextract source-filename attachments [options] AID1:outname1 [AID2:outname2 ...]`

Option	Description
`AID:outname`	Causes extraction of the attachment with the ID `AID` into the file `outname` if such an attachment exists in the source file. If the `outname` is left empty then the name of the attachment inside the source *Matroska*(tm) file is used instead. This option can be given multiple times. The attachment IDs are the same as the ones output by *mkvmerge*(1)'s `--identify` option.

2.4. Chapters extraction mode

Syntax: `mkvextract source-filename chapters [options] output-filename.xml`

Option	Description
`-s`, `--simple`	Exports the chapter information in the simple format used in the OGM tools (CHAPTER01=..., CHAPTER01NAME=...). In this mode some information has to be discarded. Default is to output the chapters in XML format.
`--simple-language Language`	If the simple format is enabled then *mkvextract*(1) will only output a single entry for each chapter atom encountered even if a chapter atom contains more than one chapter name. By default *mkvextract*(1) will use the first chapter name found for each atom regardless of its language. Using this option allows the user to determine which chapter names are output if atoms contain more than one chapter name. The `Language` parameter must be an ISO 639-1 or ISO 639-2 code.

The chapters are written to specified output file. By default the XML format understood by *mkvmerge*(1) is used. If no chapters are found in the file, the output file is not created.

2.5. Tags extraction mode

Syntax: `mkvextract source-filename tags [options] output-filename.xml`

The tags are written to specified output file in the XML format understood by *mkvmerge*(1). If no tags are found in the file, the output file is not created.

2.6. Cue sheet extraction mode

Syntax: `mkvextract` *source-filename* `cuesheet` [*options*] *output-filename.cue*

The cue sheet is written to specified output file. If no chapters or tags are found in the file, the output file is not created.

2.7. Timestamp extraction mode

Syntax: `mkvextract` *source-filename* `timestamps_v2` [*options*] *TID1:dest-filename1* [*TID2:dest-filename2* ...]

Option	Description
TID:outname	Causes extraction of the timestamps for the track with the ID *TID* into the file *outname* if such a track exists in the source file. This option can be given multiple times. The track IDs are the same as the ones output by *mkvmerge*(1)'s `--identify` option. Example: `$ mkvextract input.mkv timestamps_v2 1:ts-track1.txt 2:ts-track2.txt`

2.8. Cues extraction mode

Syntax: `mkvextract` *source-filename* `cues` [*options*] *TID1:dest-filename1* [*TID2:dest-filename2* ...]

Option	Description
TID:dest-filename	Causes extraction of the cues for the track with the ID *TID* into the file *outname* if such a track exists in the source file. This option can be given multiple times. The track IDs are the same as the ones output by *mkvmerge*(1)'s `--identify` option and not the numbers contained in the `CueTrack` element.

The format output is a simple text format: one line per `CuePoint` element with `key=value` pairs. If an optional element is not present in a `CuePoint` (e.g. `CueDuration`) then a dash will be output as the value.

Example:

```
timestamp=00:00:13.305000000 duration=- cluster_position=757741 relative_position=11
```

The possible keys are:

Option	Description
timestamp	The cue point's timestamp with nanosecond precision. The format is `HH:MM:SS.nnnnnnnnn`. This element is always set.
duration	The cue point's duration with nanosecond precision. The format is `HH:MM:SS.nnnnnnnnn`.
cluster_position	The absolute position in bytes inside the *Matroska*(tm) file where the cluster containing the referenced element starts. **Note:** Inside the *Matroska*(tm) file the `CueClusterPosition` is relative to the segment's data start offset. The value output by *mkvextract*(1)'s cue extraction mode, however, contains that offset already and is an absolute offset from the beginning of the file.

Option	Description
relative_position	The relative position in bytes inside the cluster where the `BlockGroup` or `SimpleBlock` element the cue point refers to starts. **Note:** Inside the *Matroska*(tm) file the `CueRelativePosition` is relative to the cluster's data start offset. The value output by *mkvextract*(1)'s cue extraction mode, however, is relative to the cluster's ID. The absolute position inside the file can be calculated by adding `cluster_position` and `relative_position`.

Example:

```
$ mkvextract input.mkv cues 1:cues-track1.txt 2:cues-track2.txt
```

3. Examples

Extracting both chapters and tags in their respective XML formats at the same time:

```
$ mkvextract movie.mkv chapters movie-chapters.xml tags movie-tags.xml
```

Extracting a couple of tracks and their respective timestamps at the same time:

```
$ mkvextract "Another Movie.mkv" tracks 0:video.h265 "1:main audio.aac" "2:director's comments.aac" timestamps_v2 "0:timestamps video.txt" "1:ti
```

Extracting chapters in the Ogg/OGM format and re-encoding a text subtitle track to another character set:

```
$ mkvextract "My Movie.mkv" chapters --simple "My Chapters.txt" tracks -c MS-ANSI "2:My Subtitles.srt"
```

4. Text files and character set conversions

For an in-depth discussion about how all tools in the MKVToolNix suite handle character set conversions, input/output encoding, command line encoding and console encoding please see the identically-named section in the *mkvmerge*(1) man page.

5. Output file formats

The decision about the output format is based on the track type, not on the extension used for the output file name. The following track types are supported at the moment:

Option	Description
A_AAC/MPEG2/*, A_AAC/MPEG4/*, A_AAC	All AAC files will be written into an AAC file with ADTS headers before each packet. The ADTS headers will not contain the deprecated emphasis field.
A_AC3, A_EAC3	These will be extracted to raw AC-3 files.
A_ALAC	ALAC tracks are written to CAF files.
A_DTS	These will be extracted to raw DTS files.
A_FLAC	FLAC tracks are written to raw FLAC files.
A_MPEG/L2	MPEG-1 Audio Layer II streams will be extracted to raw MP2 files.
A_MPEG/L3	These will be extracted to raw MP3 files.
A_OPUS	*Opus*(tm) tracks are written to *OggOpus*(tm) files.
A_PCM/INT/LIT, A_PCM/INT/BIG	Raw PCM data will be written to a WAV file. Big-endian integer data will be converted to little-endian data in the process.
A_REAL/*	*RealAudio*(tm) tracks are written to *RealMedia*(tm) files.
A_TRUEHD, A_MLP	These will be extracted to raw TrueHD/MLP files.
A_TTA1	*TrueAudio*(tm) tracks are written to TTA files. Please note that due to *Matroska*(tm)'s limited timestamp precision the extracted file's header will be different regarding two fields: `data_length` (the total number of samples in the file) and the CRC.
A_VORBIS	Vorbis audio will be written into an *OggVorbis*(tm) file.
A_WAVPACK4	*WavPack*(tm) tracks are written to WV files.
S_HDMV/PGS	PGS subtitles will be written as SUP files.
S_HDMV/TEXTST	TextST subtitles will be written as a special file format invented for *mkvmerge*(1) and *mkvextract*(1).
S_KATE	*Kate*(tm) streams will be written within an *Ogg*(tm) container.
S_TEXT/SSA, S_TEXT/ASS, S_SSA, S_ASS	SSA and ASS text subtitles will be written as SSA/ASS files respectively.
S_TEXT/UTF8, S_TEXT/ASCII	Simple text subtitles will be written as SRT files.
S_VOBSUB	*VobSub*(tm) subtitles will be written as SUB files along with the respective index files, as IDX files.
S_TEXT/USF	USF text subtitles will be written as USF files.
S_TEXT/WEBVTT	WebVTT text subtitles will be written as WebVTT files.
V_MPEG1, V_MPEG2	MPEG-1 and MPEG-2 video tracks will be written as MPEG elementary streams.
V_MPEG4/ISO/AVC	H.264 / AVC video tracks are written to H.264 elementary streams which can be processed further with e.g. *MP4Box*(tm) from the *GPAC*(tm) package.
V_MPEG4/ISO/HEVC	H.265 / HEVC video tracks are written to H.265 elementary streams which can be processed further with e.g. *MP4Box*(tm) from the *GPAC*(tm) package.
V_MS/VFW/FOURCC	Fixed FPS video tracks with this `CodecID` are written to AVI files.
V_REAL/*	*RealVideo*(tm) tracks are written to *RealMedia*(tm) files.
V_THEORA	*Theora*(tm) streams will be written within an *Ogg*(tm) container
V_VP8, V_VP9	VP8 / VP9 tracks are written to IVF files.
Tags	Tags are converted to a XML format. This format is the same that *mkvmerge*(1) supports for reading tags.
Attachments	Attachments are written to the output file as they are. No conversion whatsoever is done.
Chapters	Chapters are converted to a XML format. This format is the same that *mkvmerge*(1) supports for reading chapters. Alternatively a stripped-down version can be output in the simple OGM style format.
Timestamps	Timestamps are first sorted and then output as a timestamp v2 format compliant file ready to be fed to *mkvmerge*(1). The extraction to other formats (v1, v3 and v4) is not supported.

6. Exit codes

mkvextract(1) exits with one of three exit codes:

- `0` -- This exit code means that extraction has completed successfully.
- `1` -- In this case *mkvextract*(1) has output at least one warning, but extraction did continue. A warning is prefixed with the text '`Warning:`'. Depending on the issues involved the resulting files might be ok or not. The user is urged to

check both the warning and the resulting files.

- `2` -- This exit code is used after an error occurred. *mkvextract*(1) aborts right after outputting the error message. Error messages range from wrong command line arguments over read/write errors to broken files.

7. Environment variables

mkvextract(1) uses the default variables that determine the system's locale (e.g. `LANG` and the `LC_*` family). Additional variables:

Option	Description
`MKVEXTRACT_DEBUG`, `MKVTOOLNIX_DEBUG` and its short form `MTX_DEBUG`	The content is treated as if it had been passed via the --debug option.
`MKVEXTRACT_ENGAGE`, `MKVTOOLNIX_ENGAGE` and its short form `MTX_ENGAGE`	The content is treated as if it had been passed via the --engage option.

8. See also

mkvmerge(1), *mkvinfo*(1), *mkvpropedit*(1), *mkvtoolnix-gui*(1)

9. WWW

The latest version can always be found at the MKVToolNix homepage.