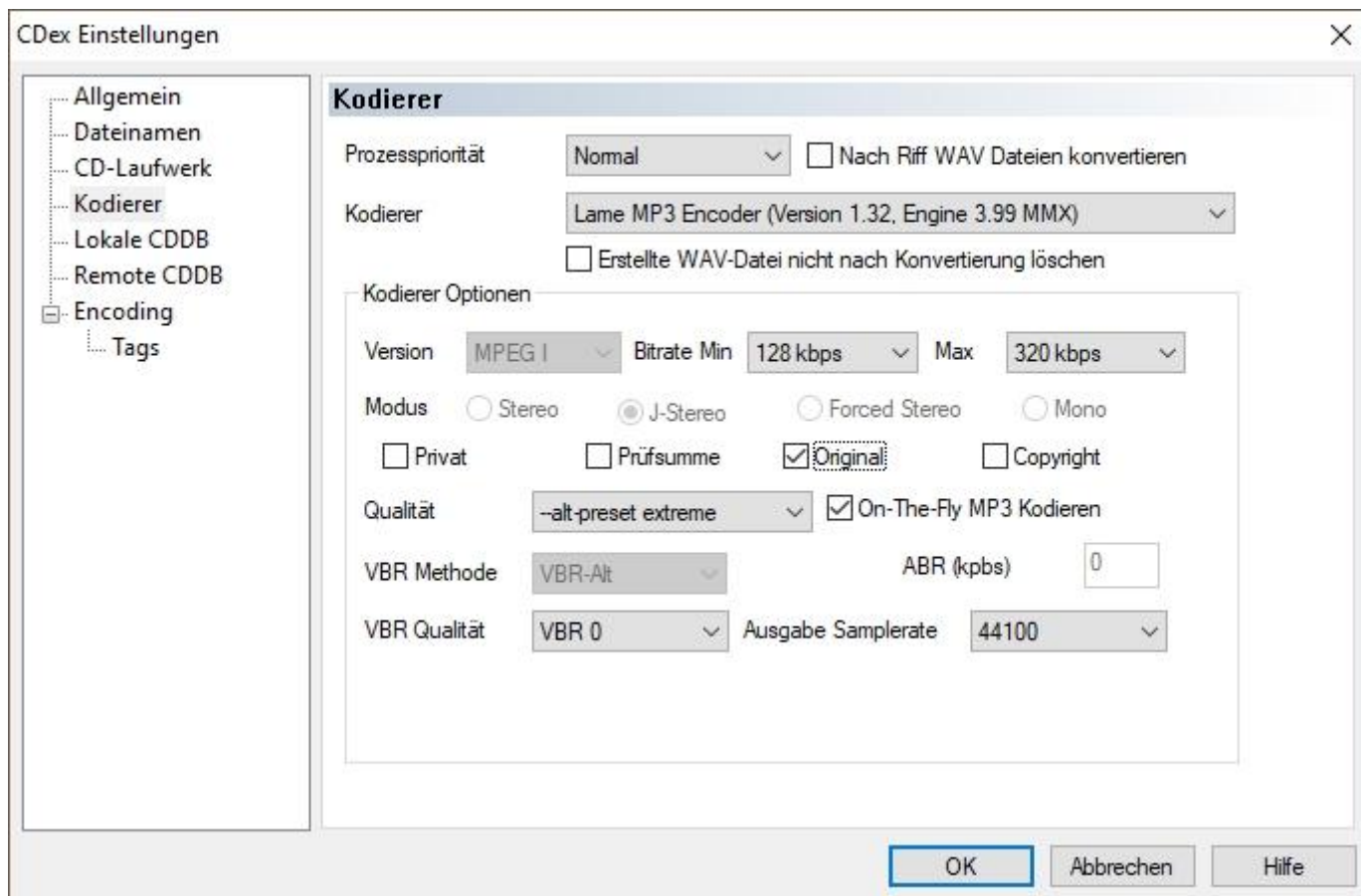


LAME - Einstellungen (am Beispiel von CDex)



Lame 3.99 Version 1.32 MMX

LAME 3.99 ist ein **Encoder**, aber "Encoder" und "Engine" werden bei LAME oft verwechselt, weil beides stimmt, nur auf unterschiedlicher Ebenen. Als **Encoder** bezeichnet man **das gesamte Programm** bzw. die Funktion, die ein Audiofile (WAV) in MP3 konvertiert. Das ist der **"LAME-Encoder 3.99"**.

Als **LAME Engine 1.32** wird **der interne Kern**, die technische Umsetzungs- und Rechenlogik innerhalb des Encoders bezeichnet. Das ist der eigentliche **Motor**, der die Psychoakustik, Bitratensteuerung uvm. berechnet. Aussen herum kann dann eine Benutzeroberfläche (wie bei CDex oder Audiograbber) sitzen. **1.32** ist die **Versionsnummer der Engine**.

MMX steht für die **interne Version des Rechenkerns** plus Hinweis auf die CPU-Optimierung. Genauer für die damalige **Intel-Prozessoren**. Die Werbung damals versprach 30% bis 40% schnellere Rechenleistung.

Zusammengefasst:

LAME 3.99 = Version des Encoders (Programm)

1.32 = Version der Encoding-Engine (Kern)

MMX = CPU-Beschleunigung

Kodierer

Früher benutzte ich LAME 3.96.1 mit --alt preset extreme. Ich entscheide mich für **Lame MP3 Encoder 3.99 in der Version 1.32 Engine 3.99 MMX**. Ich könnte über External Encoder auch die alte Version 3.96.1 von Lame einbinden.

LAME 3.99 ist die **reine Encoder-Version**, also die offizielle LAME-Version. **Version 1.32** ist nicht die LAME-Version, sondern die Engine in der Versionsnummer 1.32 der GUI / des Frontends von CDex oder Audiograbber. CDex/das Plugin/die DLL-Wrapper-Schicht = Version 1.32. Diese Frontends oder DLLs haben eigene Versionsnummern, völlig unabhängig vom LAME-Core.

MMX steht für Multimedia Extensions, eine alte Intel-Befehlserweiterung für schnellere Audio/Video-Berechnungen aus dem Pentium II/III-Zeitalter (ca. 1997). Die LAME-DLL erkennt also dass dieser CPU-Typ MMX unterstützt für schnelleres Rechnen. LAME funktionierte dadurch sogar auf uralten Pentium-II/III-PCs mit Performance-Boost.

LAME 3.99 **Encoder-Version** (relevant für die Qualität)

Version 1.32 **Engine-Version** der GUI oder LAME-DLL-Wrapper, nicht LAME selbst

MMX **MMX-Befehlssatz** des Prozessors **für schnellere Kodierung**

Die Qualität stelle ich auf **--alt preset extreme**. Dies entspricht einer VBR Qualität von **VBR 0**. Komischerweise kann ich VBR bei CDex 1.99 noch ändern. Wahrscheinlich überschreibt die Einstellung **--alt preset extreme** aber die gewählte VBR Qualität. Zur Sicherheit stelle ich auch VBR noch auf 0.

Bei der MP3-Kodierung ist VBR (Variable Bitrate) einem klassischen CBR (Constant Bitrate) in allen Qualitätsaspekten überlegen, weil es die verfügbare Datenrate intelligent einsetzt. Bei CBR wird jeder einzelne Abschnitt des Audios mit exakt gleich vielen Kilobit pro Sekunde kodiert, egal ob gerade komplexe Orchesterpassagen oder fast stille Momente vorkommen. Das bedeutet: In ruhigen Passagen wird Bandbreite verschwendet, in komplexen Passagen dagegen fehlt unter Umständen Bitrate, hörbare Artefakte können entstehen. CBR ist also technisch einfacher, aber ineffizient.

VBR analysiert das Audiomaterial in Echtzeit und gibt nur dann viele Bits aus, wenn sie wirklich benötigt werden. Ein lauter, detailreicher Refrain erhält automatisch 256-320 kbps, während eine ruhige Einleitung vielleicht nur 128-160 kbps bekommt. Immer genau so viel, wie psychoakustisch notwendig ist. Das Resultat: Hörbar bessere Qualität bei sogar kleinerer Dateigrösse. Besonders bei anspruchsvoller Musik (Klassik, Jazz, akustische Instrumente) ist der Unterschied hörbar.

Moderne Geräte und Player unterstützen VBR heute vollständig, Kompatibilitätsprobleme aus den frühen 2000er-Jahren sind praktisch verschwunden. CBR lohnt sich nur noch für Streaming-Engpässe oder uralte Hardware. Für Archivierung oder HiFi-Genuss ist VBR (V0) die bessere Wahl.

Qualität

Offizielle LAME-VBR-Skala von VBR0 (beste Qualität) bis VBR9 (schlechteste Qualität):

Stufe	Name	Ø Zielbitrate	Alte Bezeichnung	Typische Verwendung
V0	höchste Qualität	ca. 240–260 kbps	alt-preset extreme	Audiophile, Archivierung
V1	sehr hoch	ca. 225 kbps	kein alter Name	Kritisches Hören, Studiovergleich
V2	Standard	ca. 190 kbps	alt-preset standard	Empfohlen fürs Archiv
V3	gehoben	ca. 175 kbps	kein alter Name	Hochwertiges Streaming
V4	gute MP3	ca. 165 kbps	alt-preset medium	Mobilgeräte, Auto, Alltag
V5	solide	ca. 130–140 kbps	medium-fast	gute Balance für MP3-Player
V6	mittel	ca. 115 kbps	kein alter Name	Sprache, Hörbücher, Podcasts
V7	niedrig	ca. 100 kbps	kein alter Name	sehr geringe Dateigröße
V8	sehr niedrig	ca. 90 kbps	kein alter Name	Sprache in schlechter Umgebung
V9	minimale Qualität	ca. 65–85 kbps	kein alter Name	Notfälle, sehr alte Hardware

RIFF WAV Dateien

RIFF steht für **Resource Interchange File Format**. Es ist ein allgemeines Container-Format von Microsoft & IBM (1991), das entwickelt wurde, um beliebige Multimedia-Daten auf einfache Weise in Blöcken (Chunks) zu speichern. Eine WAV-Datei ist eine spezielle RIFF-Datei. Genauer steht sie am Anfang einer

klassischen WAV-Datei im Header. RIFF ist das übergeordnete Containerformat und WAV ein spezieller RIFF-Container für PCM-Audio. Darum heißen normale WAV-Dateien eigentlich RIFF/WAV.

Ich entferne die Haken und konvertiere nicht nach RIFF/WAV, weil ich keine WAV-Dateien speichern will und On The Fly lösche. On The Fly löschen bedeutet, dass die WAV-Datei schon während des Auslesens gelöscht wird.

Ausgabe Samplerate

stelle ich auf **44100 Hz**. Höhere Töne kann das menschliche Ohr sowieso nicht hören. 44100 ist Standard, niedrigere Werte wären hörbar.

Joint Stereo bei LAME

Man verbindet **Joint Stereo** sofort mit "zusammengequetscht und schlechter Klang". Das stimmt bei LAME aber nicht. **Joint Stereo bei LAME** ist ein **intelligenter Modus**, der pro Frame entscheidet, ob gerade besser echtes Stereo (L/R getrennt) oder Mid/Side (M/S) = gemeinsame Info + Differenz verwendet werden soll. Und dies automatisch und verlustfrei.

Modus	Beschreibung	Vorteil	Nachteil
Stereo	L und R immer komplett getrennt	simpel	verschenkt Bitrate, da doppelt
Joint Stereo (LAME)	entscheidet dynamisch zwischen Stereo und Mid/Side pro Musikmoment	beste Effizienz bei gleicher Qualität	keiner bei LAME
Forced Mid/Side	würde <i>immer</i> M/S erzwingen	platzsparend	kann bei Extremfällen hörbar werden

Joint Stereo bei LAME ist immer besser als normales Stereo, weil es intelligent analysiert und nur dort Bits spart, wo es garantiert verlustfrei möglich ist. Normales Stereo verschwendet Bitrate.

Im Weiteren sind noch 4 Optionen wählbar- Diese Optionen haben **keinen Einfluss auf die Audioqualität**. Es sind **Meta-Flags** im MP3-Header. Hier die Bedeutung:

Option	Bedeutung	Empfehlung
Copyright	MP3 enthält urheberrechtlich geschütztes Material	aus lassen, spielt praktisch keine Rolle, Radiostationen setzen es
Original	Markiert, dass dies das „Original“ und keine Kopie ist	ein
Privat	Frei verfügbarer, interner Nutzungs-Flag ohne definierte Funktion	immer auf aus lassen
Prüfsumme (CRC)	Fügt eine Fehlerkorrektur-Prüfsumme hinzu, ergibt etwas größere Datei	auf aus lassen, nur für alte Rundfunk-Encoder sinnvoll

Für einfach saubere und technisch korrekte MP3s: **Original aktivieren**, alles andere aus.

Unterschiede zwischen LAME 3.96.1 und LAME 3.99:

Bereich	LAME 3.96.1 (2004)	LAME 3.99 (2011)	Erklärung
Veröffentlichungsdatum	25. Juli 2004	16. Okt. 2011	3.99 ist sieben Jahre jünger und enthält viele Kompatibilitäts-/Bugfixes aus der Zwischenzeit.
VBR-Implementierung	VBR NEW ist nicht Standard, musste manuell zugeschaltet werden.	Seit 3.98 ist VBR NEW der Standard (3.99 erbt das).	Schnellere VBR-Kodierung ohne Spezialschalter ab 3.99.
Psychoakustik: CBR vs. VBR	CBR und VBR nutzen unterschiedliche Modelle.	Ab 3.99 nutzen CBR und VBR dieselbe Psychoakustik.	Einheitlicher Klangcharakter; weniger Spezialfälle/Artefakte-Unterschiede zwischen Modi.
Stabilität von VBR	Fix für seltenen VBR Bug (Crash/Datenkorruption).	Erbt die stabilisierte VBR-Pipeline; zusätzlich, viele späte Bugfixes.	3.96.1 beseitigt einen konkreten VBR NEW-Fehler, 3.99 baut darauf auf.
ID3v2/Unicode-Tags	Ältere Tags-Behandlung; weniger robust bei Unicode-Varianten.	3.99.1 behebt mehrere ID3v2-Unicode-Probleme (v. a. Big-Endian).	Bessere Kompatibilität der Tags in Playern/Bibliotheken.
VBR-Geschwindigkeit (Default)	Spürbar langsamer als spätere Versionen	Erbt die stark beschleunigte VBR aus 3.98 (Standard in 3.99).	Kürzere Encodier-Zeiten bei gleicher Zielqualität.
Verfügbare Builds	Zeittypisch primär 32-bit-Builds (häufig nur Source).	Weit verbreitet auch als x64-Builds veröffentlicht.	Bequemer Einsatz auf modernen 64-bit-Systemen.

Die große LAME-Zeitachsen-Tabelle (bis 2025, Fokus auf reale Hörqualität und Relevanz)

Skala Hörqualität = 1 (billig/Artefakte) bis 10 (audiophil/transparent/fertig)

Status 2025 = Wie man es HEUTE einordnen muss, ohne Nostalgie.

LAME-Version / Ära	Jahr	Technischer Fortschritt	Hörqualität	Historischer Einfluss	Status 2025
3.70–3.80 (Engine ~1.2x)	1999–2001	erste freie psychoakustische Engine überhaupt	3/10 erkennbar MP3, Zischlaute	Einstieg MP3 open source	veraltet
3.88 (Engine ~1.30)	2002	Fokus auf Geschwindigkeit und erste Presets	4/10	Basis für CDex/early EAC	tot
3.90.3 („ <i>Hydrogenaudio holy build</i> “)	2003	erstes „transparent“-Preset --alt-preset standard	8/10 echter Durchbruch!	erstmals audiophil anerkannt	Meilenstein
3.92–3.96	2004–2006	Finetuning, besseres Joint Stereo	8.5/10	Fanliebling Audiophile-Szene	Sweet Spot Retro
3.97	2006	modernisierte Preset-Struktur	9/10	cleaner als 3.96	historisch solide
3.98.2/3	2008	neuer psychoakustischer Layer, effizienter	9.2/10	Hydrogenaudio neuer Standard	erste vollends moderne Version
3.99.x (auch 3.99 1.32MMX)	2011	modernste VBR-Engine bis dahin, ABR stark optimiert	9.5/10	quasi-perfekt im MP3-Rahmen	bis heute extrem gültig
3.100	2017	nur Stabilitätspflege, kein hörbarer Sprung	9.5/10	letzte stabile Master	2025: de-facto Referenz
Dev-Zweig 3.101+	2020–heute	kaum Updates, MP3-Format technisch ausgereizt	9.5/10	Format selbst topt optimiert	Pflege. Kein Fortschritt mehr möglich.
Opus / AAC / FDK-AAC (nicht LAME)	2014–heute	völlig neue Generation (nicht MP3, aber relevant)	10/10 bei 128kbps	absoluter moderner Standard (Spotify, YT, Apple)	LAME überholt, aber ehrwürdig

Die ultra-kompakte Story von LAME

1999 - 2002: LAME ist ein nerdiges Open-Source-Nebenprojekt, besser als Xing oder Fraunhofer (die Erfinder des MP3-Formats), aber MP3 ist noch klar hörbar mit Zischen und Pumpen. Es tönt billig.

2003 / LAME 3.90.3 mit --alt-preset standard: Boom! Erster Moment in der Geschichte, wo MP3 plötzlich **nicht mehr vom Original unterscheidbar** ist. Die audiophile Community kippt.

2004 - 2008: Feinschliff mit den **Versionen 3.96** bis 3.98.2. MP3 wird transparent und stabil. Viele springen auf den MP3-Zug auf. Die iPods sind beliebt. Die EAC-/CDex-Ära blüht.

2011 / LAME 3.99: Technisch ausgereizt, brillante VBR-Engine, extrem effizient und **perfekt bis an die Formatgrenze**, es bleibt kein echter Raum mehr für Fortschritt.

2017 / LAME 3.100: Nur Stabilitätsabschluss, **MP3 ist am Ende seiner Optimierbarkeit angekommen.**

ab 2020: Die neuen Götter heißen **Opus und AAC**, aber LAME bleibt die ewige Meisterklasse im klassischen MP3-Format und exakt dafür bis heute Nummer 1.

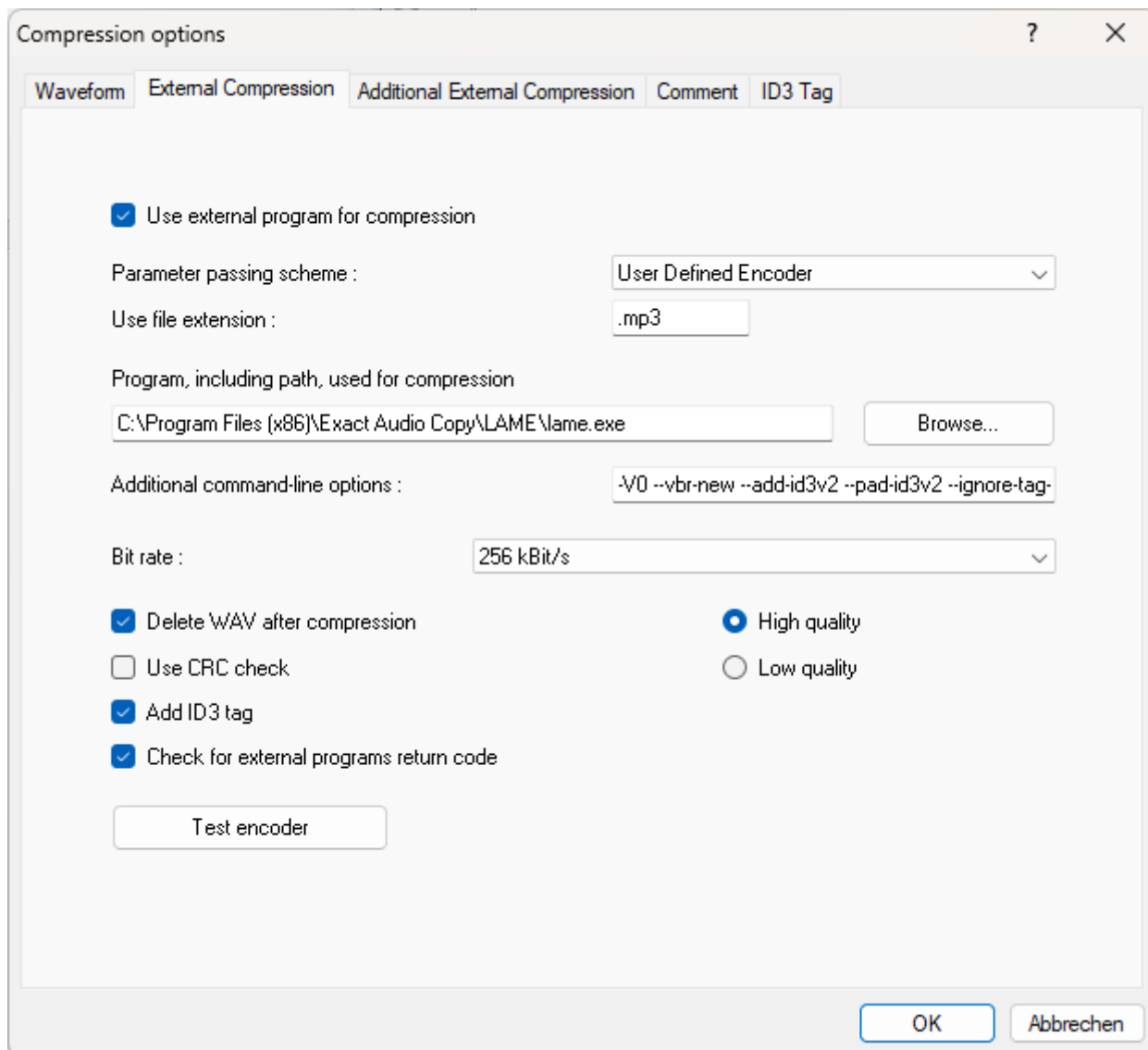
MP3 ist fertig optimiert. Opus und AAC übertreffen MP3 technisch deutlich, vor allem bei niedriger Bitrate, in der Sprachverständlichkeit und bei der Streaming-Effizienz. MP3 war Freeware und für jeden verfügbar, das gefiel vor allem Apple Music nicht wegen der Lizenz/Patent/Browser-Integration.

Plattform	Heutiges Format	Grund
Spotify	Ogg Vorbis / AAC / intern auch Opus	bessere Qualität bei gleicher Bitrate, besonders unter 192 kbps
Apple Music	AAC (Advanced Audio Codec)	AAC = offizieller MP3-Nachfolger, bessere Effizienz und lizenziert
YouTube	Opus (WebM) oder AAC (MP4)	Opus ist weltweit führend bei niedriger Bitrate, Sprache und Musik
WhatsApp, Zoom, Discord, TikTok etc.	Opus	perfekt für Sprache und Musik bei weniger als 128 kbps, minimaler Delay

LAME - Einstellungen (am Beispiel von Exact Audio Copy)

Exact Audio Copy 1.80 und LAME 3.100.1

Ich nutze den neusten und wahrscheinlich finalen Encoder **LAME 3.100.1**. So sehe ich dem MP3-File auch an, dass ich es mit EAC gerippt habe, denn früher zu CDex- und Audiograbber-Zeiten habe ich 3.96.1 oder 3.99 MMX genutzt.



Setze die Haken genau so wie oben abgebildet.

Additional command-line options :

-V0 --vbr-new --add-id3v2 --pad-id3v2 --ignore-tag-errors %source% %dest%

ist die Einstellung, die in höchster Qualität rippt, also mit variabler Bitrate. Das entspricht dem früheren `--alt -preset -extreme` oder VBR (0).

Unter "Bit rate :" werden von EAC manchmal falsche Werte eingetragen (siehe Bild oben). Wenn Du den externen Encoder (**User Defined Encoder**) angewählt hast und die **Einstellungen wie oben** gemacht hast, dann ist die **angewählte Bit Rate irrelevant**, das heisst, die Anzeige stimmt nicht! Es wird trotzdem mit variabler Bitrate gerippt und nicht wie suggerieret mit fixen 256 kBit/s.

ID3 Tags

ID3 ist der Metadaten-Standard für MP3s (Titel, Interpret, Cover usw.).

ID3v1 enthielt nur ganz wenige Informationen wie Titel, Interpret, Album, Jahr, Genre und hatte noch die Möglichkeit einen Kommentar zu speichern. ID3v1 ist vollständig in IDv2.3 und IDv2.4 aufgegangen.

Die relevanten Versionen sind **ID3v2.3** und **ID3v2.4**. Beide werden "vor" die Audiodaten des MP3-Files geschrieben und bringen keine Qualitätsunterschiede beim Audio selbst, nur bei der Kompatibilität und Funktionsbreite. Also das Wichtigste gleich voraus: **Auf die Audioqualität haben die ID3 Tags keinen Einfluss!**

Wichtigste Unterschiede

Merkmal	ID3 v2.3	ID3 v2.4
Datumfelder	Nur TYER (Jahr)	Saubere Trennung: TDRC (Aufnahmedatum), TDOR (Original), ISO-Datumsformat
Mehrere Künstler pro Feld	Nur durch manuelles Trennzeichen (zB /)	Nativ mehrere Werte pro Frame
UTF-8 Unterstützung	Nein (nur UTF-16)	Ja (UTF-8)
Cover Art	APIC	APIC
Lautstärke/ ReplayGain	Proprietär (RVAD)	standardisiert (RVA2)
Kompatibilität	perfekt auch bei alten Geräten , zB Autoradios, ältere MP3-Player	teilweise Probleme bei älteren oder billigen Geräten
Zukunftssicherheit / Norm-Treue	alt, aber "de facto Standard"	technisch richtige (modernere) Version

APIC

steht bei ID3v2-Tags für **Attached Picture**. Das ist der Frame, in dem das Coverbild bzw. eingebettete Grafik-Metadaten gespeichert werden. Darin können enthalten sein: ein **Album-Cover** (meist als JPEG oder PNG). Aber auch andere Bildtypen wie: Artist/Performer, CD-Booklet, Komponist, Live-Aufnahme-Foto, Publisher-Logo usw. Im APIC-Frame steht neben dem eigentlichen Bild auch eine Kennzeichnung, welche Art von Bild es ist (zB 3 = Front Cover). Dies wird bei Playern wie iTunes, Winamp, Foobar, Spotify etc. genutzt, um das richtige Cover anzuzeigen. Kurz: APIC = das Feld für eingebettete Bilder in einem MP3-File.

In der Praxis speichert heute zumindest kein privater Nutzer mehr das Album-Cover oder Ähnliches im MP3-File, da eine Datenbankabfrage schneller geht und weniger Speicherplatz braucht. Logitech hatte das Problem des Speicherns von Album-Covern so gelöst, dass bspw. im [Ordner Interpret-Albumtitel] das Cover auf Ebene der Songtitel als cover.jpg gespeichert wurde. Dann wusste die Server-Software, dass dies das Cover zum [Album Interpret-Albumtitel] ist.

UTF-8-Unterstützung

bedeutet, dass ein MP3-Tag oder ein Programm die Zeichenkodierung **UTF-8** versteht, also Text korrekt speichern und darstellen kann, der nicht nur aus einfachen englischen ASCII-Zeichen besteht. UTF-8 ist ein universelles Textformat, das praktisch **alle Sprachen und Sonderzeichen weltweit** darstellen kann: Umlaute: ä, ö, ü, ß oder französische Akzente: é, è, ç oder kyrillisch: Москва oder japanisch: 東京 oder Emojis: 🎧. **ID3v2.3** kann nur **UTF-16/ISO-8859-1 (Latin-1)**, also westeuropäische Zeichen. Daher gab es Probleme mit osteuropäischen, asiatischen Namen oder Emojis. **ID3v2.4** hingegen unterstützt **UTF-8**. **Viele ältere Geräte/Autoradios/MP3-Player verstehen ID3v2.4 nicht richtig** oder zeigen das Cover oder Umlaute falsch an. UTF-8-Unterstützung bedeutet also: Der MP3-Tag kann **sämtliche internationalen Zeichen korrekt anzeigen und speichern**, nicht nur **Standard-ASCII (UTF-16)**. **UTF-8 ist der moderne, globale Standard für Metadaten, Web und MP3-Tags. ASCII-kompatibel** d.h. wenige Probleme mit alten Systemen.

UTF-16-Unterstützung

Viele ältere Geräte/Autoradios/MP3-Player verstehen ID3v2.4 nicht richtig oder zeigen Sonderzeichen, das Cover oder Umlaute falsch an. **ID3v2.3 (mit UTF-16 bzw. ISO-8859-1)** ist der **"ältere Kompatibilitätsstandard"**, der auf Millionen Geräten seit den 2000er-Jahren zuverlässig läuft.

RVAD und RVA2

RVA2 ist ein ID3v2-Tag-Frame und steht für **Relative Volume Adjustment (Version 2)**. Er dient dazu, im MP3-File **Lautstärke-Informationen** zu speichern, also wie laut oder leise ein Titel im Vergleich zu einem Referenzpegel ist. Das ist wichtig, damit Musikplayer **automatisch die Lautstärke angleichen** können, ohne die Datei selbst zu verändern. Typischer Einsatzzweck: sogenannte **ReplayGain- oder Lautstärke-Normalisierung**. Player wie Foobar2000, VLC, Winamp, moderne Autoradios usw. verwenden diesen Wert mit dem Ziel Lautstärkesprünge zwischen Liedern zu vermeiden. RVA2 enthält Channel-spezifische Lautstärkekorrekturen (zB für L/R getrennt), es wird nicht hörbar in die Audiodaten eingegriffen, sondern nur als Metainformation gespeichert. RVA2 ist ein Metadatenfeld im MP3, das speichert, um wie viel dB ein Song automatisch leiser oder lauter gemacht werden soll, damit er zu anderen Lautstärke-technisch passt, ohne die Datei selbst zu verändern. Es wird von nur von **ID3v2.4 unterstützt**. **ID3v2.3** hatte dafür den älteren **"RVAD"-Frame**.

Welche Version ist nun die richtige?

Wenn du **maximale Kompatibilität** brauchst (Autoradio, ältere Geräte, alte iPods, HiFi-Streamer, USB-Sticks):

► **ID3 v2.3 mit UTF16**

Wenn dir **Zukunftssicherheit und saubere Metadaten** wichtiger sind (Streaming, Apple Music, moderne Player, Archivierung):

► **ID3 v2.4 mit UTF8**

90 % der Welt nutzt heute (2025) immer noch **ID3v2.3**, weil es einfach immer auf allen Geräten funktioniert. **ID3v2.4** ist technisch die bessere Wahl, aber es gibt noch zu viele Geräte, die es nicht korrekt lesen.

Ich verwende ID3 v2.3.

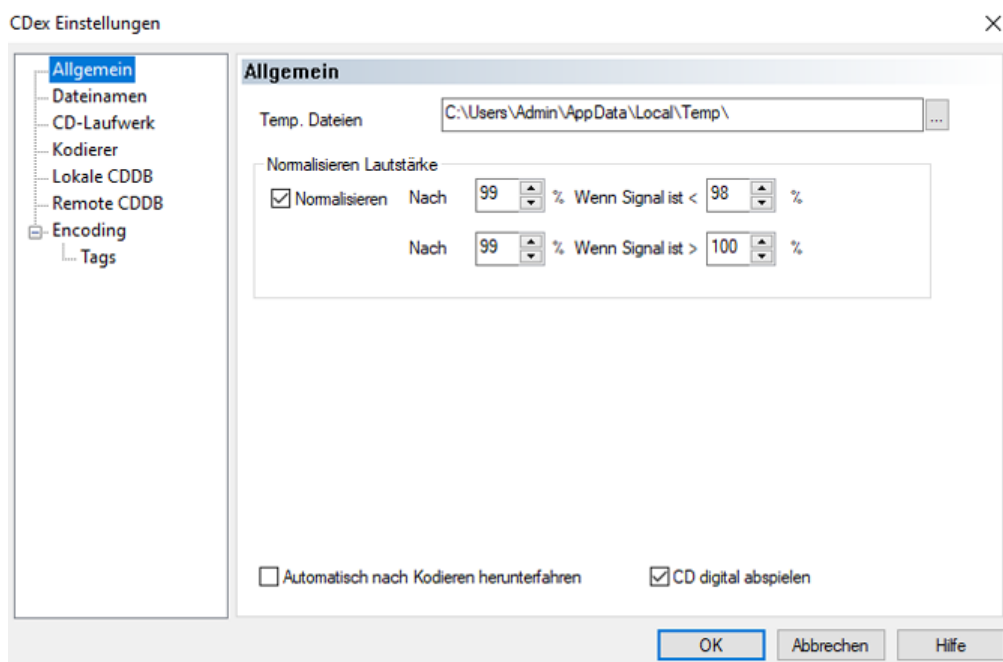
Normalisieren von MP3-Tracks

Beim MP3-Rippen ist das Normalisieren wichtig, damit alle Titel auf einem einheitlichen Lautstärke-Niveau sind.

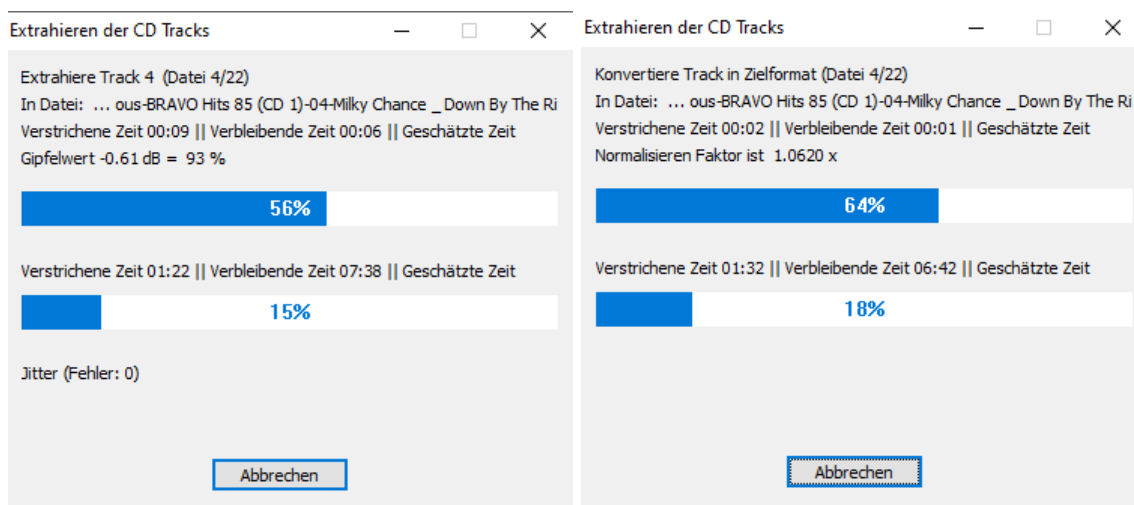
CDs und verschiedene Aufnahmen können sehr unterschiedliche Pegel haben. Ohne Normalisierung ist ein Track plötzlich sehr laut und der nächste sehr leise. Normalisieren analysiert das Audiosignal und hebt bzw. senkt den Pegel so an, dass der lauteste Punkt einen definierten Maximalwert erreicht. Im Beispiel von CDEX unten wird auf 99% normalisiert. Dadurch klingt das Album oder die Playlist konsistent, ohne hörbare Verzerrungen oder Clipping.

Kurz gesagt: **Normalisieren sorgt für gleichbleibende Lautstärke und verhindert Übersteuerung.**

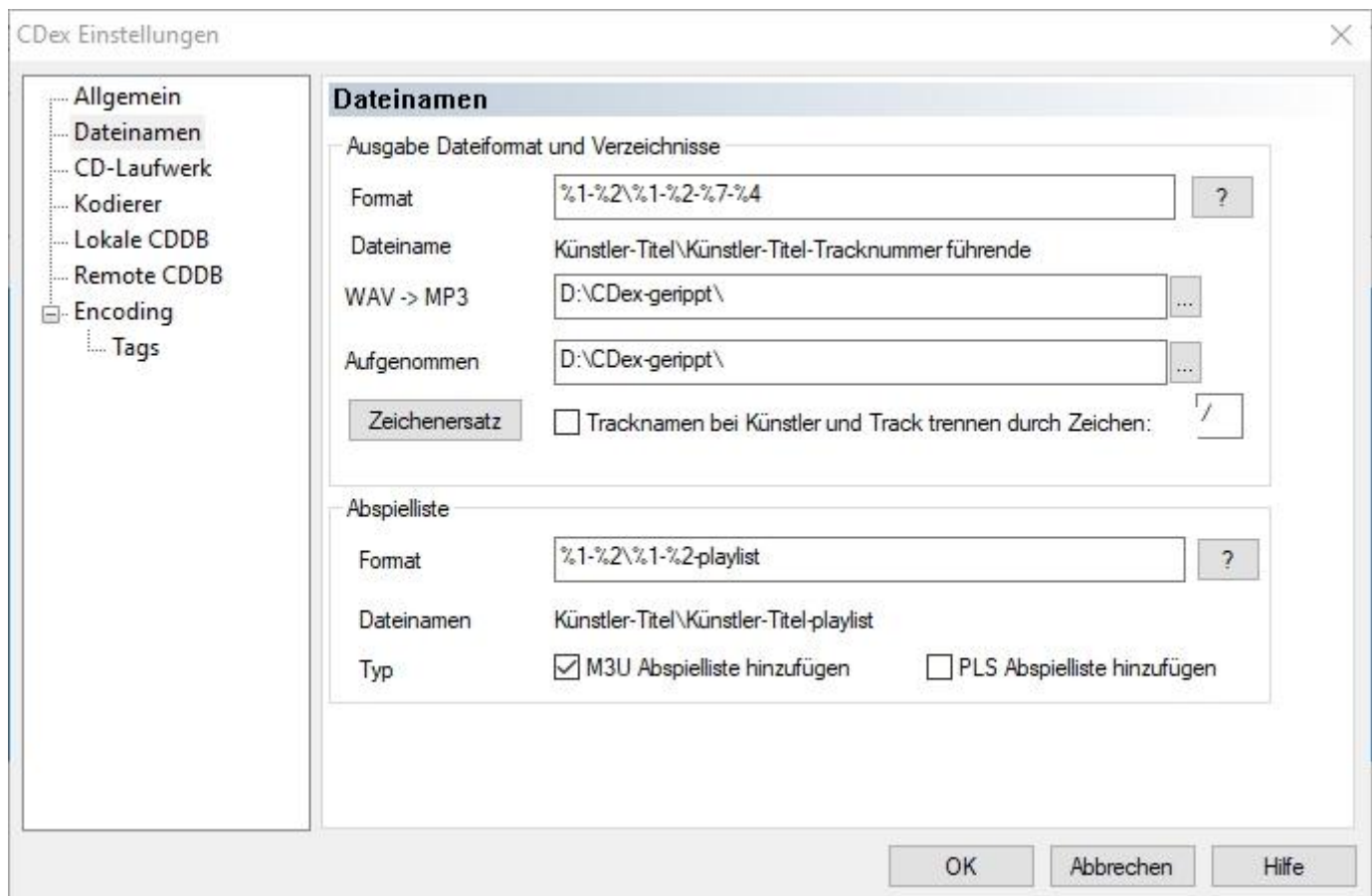
Ich habe absichtlich die Grenzen 98% und 100% gewählt, damit nicht jeder Track auf der CD auch normalisiert werden muss. Die meisten Tracks sollten schon richtig abgemischt sein.



Vor allem Sampler-Alben haben manchmal Tracks mit sehr unterschiedlichen Lautstärken. Dieser Beispiel-Track hatte einen Gipfelwert von 93% und wurde mit dem Faktor 1.062 normalisiert. Somit ergibt sich eine maximale Lautstärke dieses Tracks von 99%.



Playlists M3U (am Beispiel von CDex)



Eine **Playlist** ist eigentlich nur eine **Textdatei**, die eine Liste von Audiodateien enthält, damit ein Player weiss, welche Titel in welcher Reihenfolge abgespielt werden sollen. Der Unterschied liegt im Format (Syntax), nicht in der Funktion.

Ich beschrifte die MP3-Dateien wie folgt, indem ich in meiner **Playlist** folgendes Format verwende:

%1-%2\\%1-%2-%7-%4

%1 = Künstler/Interpret/Band

%2 = Albumname

%7 = Tracknummer

%4 = Titel des Songs

CDex kennt noch weitere Codes unter -> CDex Einstellungen -> Dateinamen.

Das ergibt dann beispielsweise folgende Datei (**Playlist**) im **Format M3U**

Springsteen Bruce-Born In The USA-01-Born In The USA.mp3

Springsteen Bruce-Born In The USA-02- Cover Me.mp3

Springsteen Bruce-Born In The USA-03-Darlington County.mp3

...

Springsteen Bruce-Born In The USA-12-My Hometown.mp3

Ich kann sie nachträglich auch noch leicht in einem Editor bearbeiten, da es sich um eine reine *.txt Text-Datei handelt mit der Datei-Endung *.m3u.

Ich generiere nur die **Playlist** im **Format M3U** und nenne sie der Einfachheit halber gleich mit dem Künstler und Albumnamen: **Springsteen Bruce-Born In The USA.m3u**

*.m3u **M3U** und *.pls **PLS** sind "Erfindungen" von **WINAMP**, welches das Abspielen von MP3s auf Computern revolutionierte. M3U ist das simpelste und uralte Basis-Format. PLS ist strukturierter. Heute ist

*.m3u8 **M3U8** der wichtigste moderne Nachfolger (UTF-8 fähig, streamingtauglich). Welche man benutzt, hängt vom Player ab. Viele können mehrere lesen.

Eine **Playlist** im **Format PLS** sieht ganz ähnlich aus:

```
[playlist]
NumberOfEntries=12
Version=2
File1=Springsteen Bruce-Born In The USA-01-Born In The USA.mp3
File2=Springsteen Bruce-Born In The USA-02- Cover Me.mp3
File3=Springsteen Bruce-Born In The USA-03-Darlington County.mp3
...
Springsteen Bruce-Born In The USA-12-My Hometown.mp3
```

.m3u

- **sehr einfaches, minimalistisches Format**
- ursprünglich für **WINAMP** (Frühzeit der MP3-Player)
- enthält in der Basisversion einfach pro Zeile einen Dateipfad
- optional erweiterbar als **M3U-Extended** (#EXTM3U, #EXTINF: für Titel, Länge, Interpret etc.)
- UTF-8-Variante = .m3u8 M3U8 ist heute Standard bei Streaming (zB HLS von Apple)

.pls

- ebenfalls von **WINAMP**, aber etwas strukturierter
- INI-ähnlicher Aufbau (nummerierte Felder)
- unterstützt Metadaten explizit (Title, Length)
- vor allem bei Internetradio und früher Shoutcast beliebt

Heute gibt es natürlich weitere, modernere Varianten, hier die Wichtigsten:

.m3u8 M3U8 heutiger de-fakto Standard bei Streaming / HLS

.xspf XSPF (XML Shareable Playlist Format, open-source Standard)

.wpl WPL Microsoft Windows Media Player

.cue CUE eher CD-Struktur zum Indexieren in einer einzigen großen MP3-Datei

So sieht die moderne **Playlist** im **Format M3U8** in der Streaming-Variante aus:

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:300
#EXT-X-MEDIA-SEQUENCE:0
#EXTINF:279,01 - Born In The USA.mp3
#EXTINF:278,02 - Cover Me.mp3
#EXTINF:252,03 - Darlington County.mp3
...
#EXTINF:296,12 - My Hometown.mp3
#EXT-X-ENDLIST
```

Hier sind die **wichtigsten M3U bzw. M3U8-Steuerbefehle**
(also Zeilen mit #, die mehr sind als nur Dateipfade) mit kurzer Erklärung:

Befehl	Bedeutung
#EXTM3U	Signalisiert: <i>Extended M3U-Format</i> . Muss immer ganz oben stehen, falls Metadaten folgen
#CURTRACK	Aktuelle Tracknummer, wenn Playlist abgebrochen wird. Steht bei einigen Playern noch vor #EXTM3U
#EXTINF:<Dauer>,<Titel>	Metadaten zum folgenden Track: <Dauer in Sekunden>, danach <Titel/Interpret>. Der nächste Eintrag ist dann der Pfad zur Datei
#EXTALB:<Albumname>	inoffiziell, von manchen Playern verwendet - Albumtitel der gesamten Playlist
#EXTART:<Artist>	inoffiziell - Künstler der Playlist. Wird meist ignoriert
#PLAYLIST:<Name>	inoffiziell - Name der Playlist selbst. Wird selten ausgewertet
#EXTGRP:<Gruppe>	Gruppieren Tracks (Streaming / TV-Channel-Gruppierung, zB SPORT)
#EXTVLCOPT:<Option>	Spezielle VLC-Option für den folgenden Track (wie Sprache, Buffer usw.) nur VLC-spezifisch

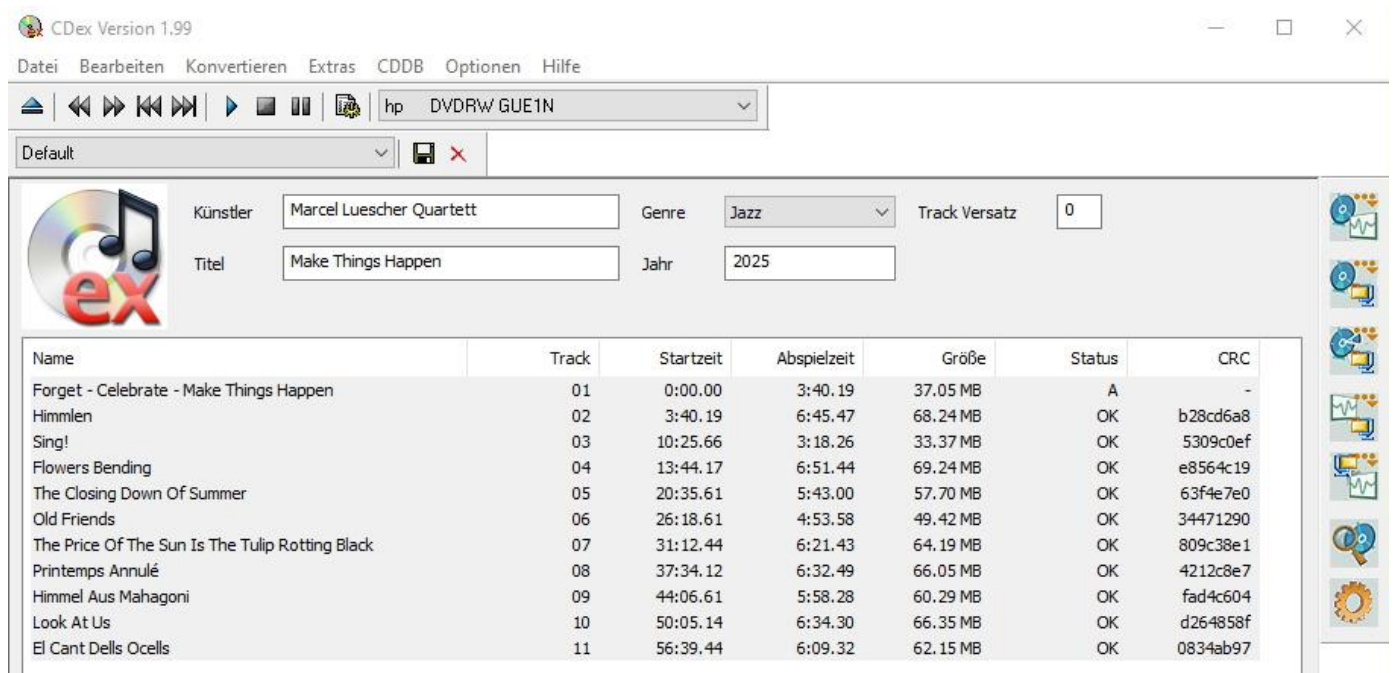
Speziell für Streaming / HLS-kompatible .m3u8 (Apple-Standard)

Befehl	Bedeutung
#EXT-X-VERSION:<Nummer>	Version des HLS-Standards (3 oder 4 am häufigsten)
#EXT-X-TARGETDURATION:<Sekunden>	Länge des längsten Segments - für Streaming-Puffer
#EXT-X-MEDIA-SEQUENCE:<Nummer>	Startpunkt der Sequenz - wichtig bei Livestreams
#EXT-X-STREAM-INF:	Gibt an: folgt ein alternativer Quality-Stream / Bitrate (Adaptive Streaming)
#EXT-X-ENDLIST	Signalisiert: Playlist ist abgeschlossen (keine Livestream-Fortsetzung)

Fehler beim CD rippen

Die CD wird mit dem 6. Button rechts von oben ausgelesen: **Album Information von Remote CDDB lesen**.

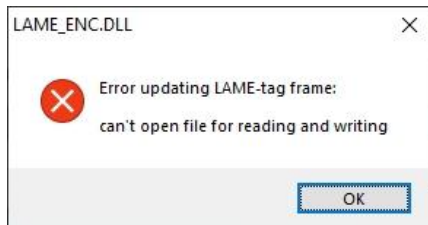
	CD Track(s) nach WAV-Datei extrahieren
	CD Tracks(s) nach komprimierter Audio-Datei extrahieren
	Einen Teil einer CD nach WAV oder komprimierter Audio-Datei extrahieren
	WAV Dateien nach komprimierte Audio-Datei extrahieren
	Komprimierte Audio-Datei zurück nach WAV Datei konvertieren
	Album Information von Remote CDDB lesen
	Konfigurieren (CDex Einstellungen)



Das Rippen wird über den 2. Button rechts von oben gestartet: **CD Tracks(s) nach komprimierter Audio-Datei extrahieren**.

Unter anderem wird auch der *Track 08 Printemps Annulé* generiert. Das Auslesen mit **ID3 Tag Version 2.3** (mit UTF-16) kann Probleme machen. Die Version 2.4 (UTF-8) führt nicht zu Problemen. Warum ich trotzdem die Version 2.3 verwende, wird im *Kapitel ID3 Tags* erklärt.

Verwende für Künstler und Titel keine Sonderzeichen wie "ä", "ö" oder "ü". Denn daraus werden die MP3-Dateien generiert. Windows hat mit gewissen Sonderzeichen Mühe und generiert dann komische Dateinamen. Das Sonderzeichen "é" im Track 8 Printemps Annulé führte dazu, dass die MP3-Informationen nicht richtig geschrieben wurden. Manchmal werden Sonderzeichen aber richtig konvertiert. CDex ist da inkonsequent.



Diese Fehlermeldung erscheint, wenn CDex MP3-Informationen nicht richtig schreiben kann. Dies passiert hier bspw. wegen **Track Nummer 8** und dem Sonderzeichen **"é"**. **Das MP3-File wird trotzdem generiert.** Das ist wichtig zu wissen. **Das Rippen nicht abbrechen.** Sondern nur auf **OK** klicken.



MP3TAG Der universelle Tag Editor.

Mit dem Programm MP3Tag können dann fehlerhafte MP3-Tags richtiggestellt werden.

Also zusammengefasst: Vor allem für die **Tags von Künstler und Titel auf keinen Fall Sonderzeichen verwenden.** Für Deutsch empfiehlt sich statt "ä" = ae, statt "ö" = oe und statt "ü" ue zu nutzen. Auch "&", "é", "*", "+", "\$" nur sparsam verwenden, denn mit der Einstellung ID3 Tag auf Version 2.3 kann es Probleme beim Schreiben der ID3 Tags geben. Auch keine Steuerzeichen wie "/" oder ">" oder "\" im Titelnamen, da das auch zu Ärger führen könnte.

Wenn die ID3 Tags Probleme bereiten, liegt es meistens an den verwendeten Sonderzeichen.

CDs mit Kopierschutz

Manchmal gelingt das Rippen nicht oder bricht ab oder die Tracks klingen schrecklich mit elektronischem "Jitter-Rauschen".

Dann kann das daran liegen, dass die CD selbst defekt oder zu stark zerkratzt ist. Oder eben, sie enthält einen Kopierschutz. Als das CD-Kopieren gross in Mode kam Ende der 90er-Jahre wollte die Musik-Industrie ihre Käufe schützen. Wenn man einen CD-Player hatte, der gewisse CDs dann einfach nicht abspielen konnte, war das einfach nur ärgerlich. Und ich meine jetzt nicht das Laufwerk im PC, sondern den Player für die Stereo-Anlage.

Profis spielten die CDs einfach digital ab auf einem Player, der das konnte, und nahmen die Musik als eine grosse WAV-Datei über LINE-IN ihrer Soundkarte auf. Ich könnte wetten, dass damals von den kopiergeschützten CDs sogar mehr Kopien in Umlauf kamen, weil es eine Herausforderung wurde, eben diese CDs zu kopieren.

Diese 4 CDs sind bspw. weder mit CDex noch mit Exact Audio Copy "rippbar", **Kopierschutz:**

- Tracy Chapman - Tracy Chapman (Reissue 2001–2003) **Cactus Data Shield 200 / Copy Control**
- Jewel - Goodbye Alice in Wonderland (EU 2006 / EMI) **Cactus Data Shield 300 / Copy Control (EMI)**
- Sasha - Greatest Hits (Warner 2006) **Key2Audio² (Sony/Warner Lizenz)**
- Wolfsheim – Casting Shadows (2003, EMI/Motor) **Cactus Data Shield 200 / Copy Control**